

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-208627

(43)Date of publication of application : 25.07.2003

(51)Int.Cl.

G06T 11/00  
G06F 15/16  
G06F 15/167  
G06T 1/20

(21)Application number : 2002-006393

(71)Applicant : SONY CORP

(22)Date of filing : 15.01.2002

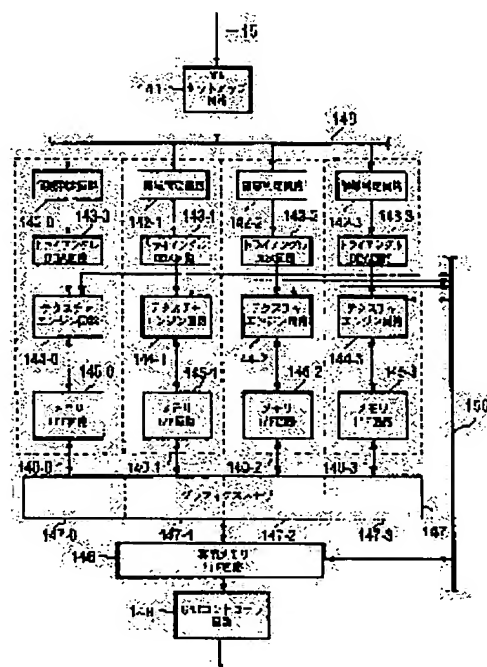
(72)Inventor : INADA TETSUGO  
SATO HITOSHI  
YAMAGUCHI YUJI

## (54) IMAGE PROCESSING DEVICE AND METHOD THEREFOR

## (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide an image processing device and a method therefor capable of performing parallel processing irrespective of operation mode, realizing parallel processing having high execution efficiency without receiving the restriction for the number of pixels capable of being parallel processed, and improving the performance.

**SOLUTION:** The same triangle data are transmitted to each processing module 140-0 to 140-3 in parallel, and each processing module 140-0 to 140-3 receives the information related to the same triangle to discriminate whether it is included in regions RGN0 to RGN3 in charge of own module or not. Each processing module has region discrimination circuits 142-0 to 142-3 for outputting the input information to DDA circuits 143-0 to 143-3 if it is included in the regions RGN0 to RGN3, draws only pixel included in the regions RGN0 to RGN3 in charge of own module, and transmits it to corresponding memory modules 147-0 to 147-3.



## LEGAL STATUS

[Date of request for examination]

29.11.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2003-208627

(P2003-208627A)

(43)公開日 平成15年7月25日(2003.7.25)

(51)Int.Cl. <sup>7</sup>	識別記号	F I	テマコード*(参考)
G 0 6 T 11/00	1 0 0	G 0 6 T 11/00	1 0 0 A 5 B 0 4 5
G 0 6 F 15/16	6 1 0	G 0 6 F 15/16	6 1 0 A 5 B 0 5 7
15/167		15/167	C 5 B 0 8 0
G 0 6 T 1/20		G 0 6 T 1/20	B

審査請求 未請求 請求項の数31 O L (全 25 頁)

(21)出願番号 特願2002-6393(P2002-6393)

(22)出願日 平成14年1月15日(2002.1.15)

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 稲田 徹悟

東京都品川区東五反田1丁目14番10号 株式会社ソニー木原研究所内

(72)発明者 佐藤 仁

東京都品川区東五反田1丁目14番10号 株式会社ソニー木原研究所内

(74)代理人 100094053

弁理士 佐藤 隆久

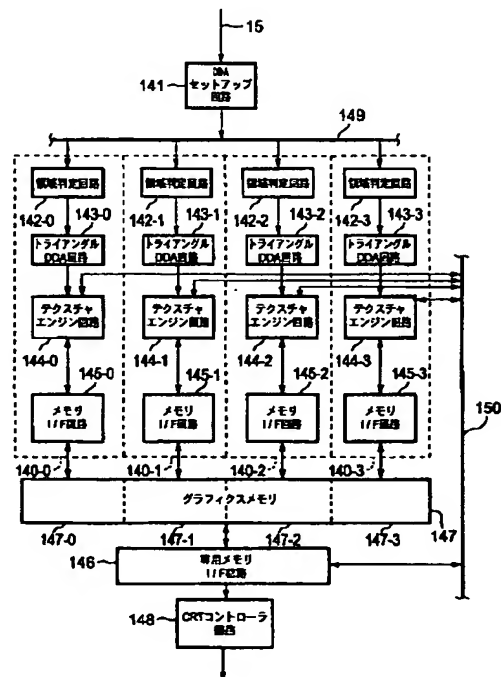
最終頁に続く

(54)【発明の名称】 画像処理装置およびその方法

(57)【要約】

【課題】動作モードにかかわらず並列処理を行うことができ、並列処理できるピクセル数に対する制約を受けずに実行効率の高い並列処理を実現でき、ひいては、性能の向上を図ることができる画像処理装置およびその方法を提供する。

【解決手段】同じ三角形データを各処理モジュール140-0~140-3に並列に送り、各処理モジュール140-0~140-3は、同一の三角形に関する情報を受けて、自モジュールの担当領域RGN0~RGN3に含まれるか否かを判断し、含まれる場合には入力情報をDDA回路143-0~143-3に出力する領域判定回路142-0~142-3を有し、自モジュールの担当する領域RGN0~RGN3に含まれるピクセルのみを描画して対応するメモリモジュール147-0~147-3に送る。



## 【特許請求の範囲】

【請求項 1】 複数の処理モジュールが処理データを共有して並列処理を行いメモリに描画する画像処理装置であって、

上記複数の処理モジュールはそれぞれ、描画すべきプリミティブに関する情報を受けて、プリミティブがあらかじめインターリーブ状に分割された自モジュールの担当領域に含まれるか否かを判定する領域判定回路と、

上記領域判定回路により自モジュールの担当領域に含まれると判定されると、入力情報に基づいて担当領域に対する描画処理を行う処理回路と、を含む画像処理装置。

【請求項 2】 上記複数の処理モジュールの領域判定回路には、描画すべきプリミティブに関する同一の情報がブロードキャストされる請求項 1 記載の画像処理装置。

【請求項 3】 上記複数の処理モジュールの領域判定回路には、描画すべき異なるプリミティブに関する情報がそれぞれ供給される請求項 1 記載の画像処理装置。

【請求項 4】 上記複数の処理モジュールの処理回路は、自モジュールの担当する領域に含まれるピクセルのみを描画する請求項 1 記載の画像処理装置。

【請求項 5】 上記複数の処理モジュールの処理回路は、自モジュールの担当する領域に含まれるピクセルのみを描画する請求項 2 記載の画像処理装置。

【請求項 6】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 4 記載の画像処理装置。

【請求項 7】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 5 記載の画像処理装置。

【請求項 8】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 4 記載の画像処理装置。

【請求項 9】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 5 記載の画像処理装置。

【請求項 10】 上記メモリはメモリインターリーブされた複数のメモリモジュールを含み、

上記複数の処理モジュールと上記複数のメモリモジュールとが一对一に対応しており、

上記複数の処理モジュールの処理回路は、自モジュールの担当する領域に対する処理結果のみを描画して対応するメモリモジュールに出力する請求項 1 記載の画像処理装置。

【請求項 11】 上記メモリはメモリインターリーブされた複数のメモリモジュールを含み、

上記複数の処理モジュールと上記複数のメモリモジュールとが一对一に対応しており、

上記複数の処理モジュールの処理回路は、自モジュールの担当する領域に含まれるピクセルのみを描画して対応するメモリモジュールに出力する請求項 2 記載の画像処理装置。

【請求項 12】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 10 記載の画像処理装置。

【請求項 13】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 11 記載の画像処理装置。

【請求項 14】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 10 記載の画像処理装置。

【請求項 15】 各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、

上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 11 記載の画像処理装置。

【請求項 16】 上記領域判定回路は、プリミティブが自モジュールの担当領域に含まれると判定した場合には入力情報を上記処理回路に出力し、含まれていないと判定した場合には入力情報を上記処理回路に出力せず棄てる請求項 1 記載の画像処理装置。

【請求項 17】 複数の処理モジュールが処理データを共有して並列処理を行いメモリに描画する画像処理方法であって、

上記複数の処理モジュールに対してインターリーブ状に分割して自モジュールが処理する担当領域を割り当てておき、

上記複数の処理モジュールに対して描画すべきプリミティブに関する情報を供給し、

各処理モジュールにおいて、プリミティブがあらかじめインターリーブ状に分割された自モジュールの担当領域に含まれるか否かを判定し、

自モジュールの担当領域に含まれると判定した場合に、供給された情報に基づいて担当領域に対する描画処理を行う画像処理方法。

【請求項 18】 上記複数の処理モジュールには、描画すべきプリミティブに関する同一の情報をブロードキャストする請求項 17 記載の画像処理方法。

【請求項 19】 上記複数の処理モジュールには、描画すべき異なるプリミティブに関する情報をそれぞれ供給する請求項 17 記載の画像処理方法。

【請求項 20】 上記複数の処理モジュールにおいては、自モジュールの担当する領域に含まれるピクセルのみを描画する請求項 17 記載の画像処理方法。

【請求項 21】 上記複数の処理モジュールにおいては、自モジュールの担当する領域に含まれるピクセルのみを描画する請求項 18 記載の画像処理方法。

【請求項 22】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 20 記載の画像処理方法。

【請求項 23】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 21 記載の画像処理方法。

【請求項 24】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 20 記載の画像処理方法。

【請求項 25】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 21 記載の画像処理方法。

【請求項 26】 上記メモリを、複数のメモリモジュール

にインターリーブして、上記複数の処理モジュールと上記複数のメモリモジュールとを一对一に対応させ、上記複数の処理モジュールにおいては、自モジュールの担当する領域に対する処理結果のみを描画して対応するメモリモジュールに出力する請求項 17 記載の画像処理方法。

【請求項 27】 上記メモリを、複数のメモリモジュールにインターリーブして、上記複数の処理モジュールと上記複数のメモリモジュールとを一对一に対応させ、上記複数の処理モジュールにおいては、自モジュールの担当する領域に対する処理結果のみを描画して対応するメモリモジュールに出力する請求項 18 記載の画像処理方法。

【請求項 28】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 26 記載の画像処理方法。

【請求項 29】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンしながらピクセルを発生する請求項 27 記載の画像処理方法。

【請求項 30】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 26 記載の画像処理方法。

【請求項 31】 各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールとしては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する請求項 27 記載の画像処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、複数の演算処理モジュールが処理データを共有して並列処理を行う画像処理装置およびその方法に関するものである。

【0002】

【従来の技術】近年、3次元コンピュータグラフィックス (3D Computer Graphics) をハードウェアで高速に実行するグラフィックス LSI の普及は著しく、特にゲーム機やパーソナルコンピュータ (PC) では、このグラフィックス LSI を標準で搭載

しているものが多い。

【0003】図1は、グラフィックス描画を行う一般的な画像処理装置を概念的に示す図である。

【0004】この画像処理装置1は、ピクセル発生部2、および複数個(図1の例では4個)のメモリモジュール3-0~3-3を有している。この画像処理装置1では、ピクセル発生部2において、単位図形たとえば三角形プリミティブPMがラスタライズされて複数のピクセルPXが生成される。そして、生成されたピクセルPXが図示しない接続回路を介して複数のメモリモジュール3-0~3-3に送られる。このとき、生成されたピクセルは、ピクセルのXY座標に応じて対応するメモリモジュールに送られる。

【0005】このように、図1の画像処理装置1は、基本的に一つのピクセル発生部2で発生した複数のピクセルをインターリーブされた複数のメモリモジュール3-0~3-3に分配する。

【0006】

【発明が解決しようとする課題】ところで、グラフィックスLSIにおける技術的進歩は早く、「Direct X」で採用された「Vertex Shader」や「Pixel Shader」に代表される機能面での拡張が続けられているとともに、CPUを上回るペースで性能の向上が望まれている。

【0007】グラフィックスLSIの性能を向上させるには、LSIの動作周波数を上げることで一応の対応は可能である。ところが、上述した画像処理装置1は、1系統のピクセル発生部2で発生した複数のピクセルを複数のメモリモジュールに分配する基本構成を有することから、LSIの動作周波数を上げるだけでは、満足した性能を得ることはできない。

【0008】グラフィックスLSIの性能を向上させるには、LSIの動作周波数を上げるだけではなく、並列処理の手法を利用することが有効である。並列処理の手法を大別すると以下ようになる。第1は領域分割による並列処理法であり、第2はピクセルレベルでの並列処理法である。

【0009】上記分類は並列処理の粒度に基づいており、領域分割並列処理の粒度が粗く、ピクセルレベル並列処理の粒度が細かい。それぞれの手法の概要を以下に述べる。

#### 【0010】領域分割による並列処理

領域分割による並列処理は、画面を複数の矩形領域に分割し、複数の処理モジュールそれぞれが担当する領域を割り当てながら並列処理する手法である。

#### 【0011】ピクセルレベルでの並列処理

ピクセルレベルでの並列処理は、粒度の細かい並列処理の手法である。図2は、ピクセルレベルでの並列処理の手法に基づくプリミティブレベルでの並列化処理について概念的に示す図である。図2のように、ピクセルレベ

ルでの並列処理の手法では三角形をラスタライズする際に、 $2 \times 8$ のマトリクス状に配列されたピクセルからなるピクセルスタンプ(Pixel Stamp)PSと呼ばれる矩形領域単位にピクセルが生成される。図2の例では、ピクセルスタンプPS0からピクセルスタンプPS7までの合計8個のピクセルスタンプが生成されている。これらピクセルスタンプPS0~PS7に含まれる最大16個のピクセルが同時に処理される。この手法は、粒度が細かい分、並列処理の効率が良い。

【0012】しかしながら、上述した領域分割による並列処理の場合、各処理モジュールを効率良く並列動作させるためには、各領域に描画されるべきオブジェクトをあらかじめ分類する必要があり、シーンデータ解析の負荷が重い。また、1フレーム分のシーンデータが全て揃った上で描画を開始するのではなく、オブジェクトデータが与えられると即描画を開始するいわゆるイミディエートモードでの描画を行う際には並列性を引き出すことができない。

【0013】また、ピクセルレベルでの並列処理の場合、グラフィックスで処理されるプリミティブの大きさは小さくなる傾向にあることから、ピクセルスタンプのサイズを大きくしても無効なピクセルが増えるだけで実行効率が上がらない。したがって、この手法によって並列処理できるピクセルの数には限界がある。

【0014】本発明は、かかる事情に鑑みてなされたものであり、その目的、複数の処理モジュールが処理データを共有して並列処理する際に、動作モードにかかわらず並列処理を行うことができ、並列処理できるピクセル数に対する制約を受けずに実行効率の高い並列処理を実現でき、ひいては、性能の向上を図ることができる画像処理装置およびその方法を提供することにある。

【0015】

【課題を解決するための手段】上記目的を達成するため、本発明の第1の観点は、複数の処理モジュールが処理データを共有して並列処理を行いメモリに描画する画像処理装置であって、上記複数の処理モジュールはそれぞれ、描画すべきプリミティブに関する情報を受けて、プリミティブがあらかじめインターリーブ状に分割された自モジュールの担当領域に含まれるか否かを判定する領域判定回路と、上記領域判定回路により自モジュールの担当領域に含まれると判定されると、入力情報に基づいて担当領域に対する描画処理を行う処理回路とを含む。

【0016】第1の観点では、上記複数の処理モジュールの領域判定回路には、描画すべきプリミティブに関する同一の情報がブロードキャストされる。

【0017】また、第1の観点では、上記複数の処理モジュールの領域判定回路には、描画すべき異なるプリミティブに関する情報がそれぞれ供給される。

【0018】第1の観点では、好適には、上記複数の処

理モジュールの処理回路は、自モジュールの担当する領域に含まれるピクセルのみを描画する。

【0019】また、第1の観点では、各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンしながらピクセルを発生する。

【0020】また、第1の観点では、各処理モジュールの担当領域は、所定の単位領域を上記処理モジュールの数だけ分割した各領域がそれぞれ割り当てられ、上記複数の処理モジュールの処理回路は、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する。

【0021】また、第1の観点では、上記メモリはメモリインターリーブされた複数のメモリモジュールを含み、上記複数の処理モジュールと上記複数のメモリモジュールとが一对一に対応しており、上記複数の処理モジュールの処理回路は、自モジュールの担当する領域に対する処理結果のみを描画して対応するメモリモジュールに出力する。

【0022】また、好適には、上記領域判定回路は、プリミティブが自モジュールの担当領域に含まれると判定した場合には入力情報を上記処理回路に出力し、含まれていないと判定した場合には入力情報を上記処理回路に出力せず棄てる。

【0023】本発明の第2の観点は、複数の処理モジュールが処理データを共有して並列処理を行いメモリに描画する画像処理方法であって、上記複数の処理モジュールに対してインターリーブ状に分割して自モジュールが処理する担当領域を割り当ておき、上記複数の処理モジュールに対して描画すべきプリミティブに関する情報を供給し、各処理モジュールにおいて、プリミティブがあらかじめインターリーブ状に分割された自モジュールの担当領域に含まれるか否かを判定し、自モジュールの担当領域に含まれると判定した場合に、供給された情報に基づいて担当領域に対する描画処理を行う。

【0024】第2の観点では、上記複数の処理モジュールには、描画すべきプリミティブに関する同一の情報をブロードキャストする。

【0025】第2の観点では、上記複数の処理モジュールには、描画すべき異なるプリミティブに関する情報をそれぞれ供給する。

【0026】好適には、上記複数の処理モジュールにおいては、自モジュールの担当する領域に含まれるピクセルのみを描画する。

【0027】また、第2の観点では、各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンしながらピクセルを発生する。

【0028】また、第2の観点では、各処理モジュールの担当領域として、所定の単位領域を上記処理モジュールの数だけ分割した各領域をそれぞれ割り当て、上記複数の処理モジュールにおいては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する。

【0029】また、第2の観点では、上記メモリを、複数のメモリモジュールにインターリーブして、上記複数の処理モジュールと上記複数のメモリモジュールとを一对一に対応させ、上記複数の処理モジュールにおいては、自モジュールの担当する領域に対する処理結果のみを描画して対応するメモリモジュールに出力する。

【0030】本発明によれば、たとえば同じ描画プリミティブに関する情報が各処理モジュールにブロードキャストされる。また、各処理モジュールは、インターリーブされた各メモリモジュールと対応付けられる。そして、各処理モジュールの領域判定回路において、入力情報に係るプリミティブがあらかじめインターリーブ状に分割された自モジュールの担当領域に含まれるか否かの判定が行われる。各領域判定回路において、プリミティブが自モジュールの担当領域に含まれると判定した場合には入力情報が自モジュールの処理回路に出力される。一方、含まれていないと判定した場合には入力情報が処理回路に出力されず破棄される。そして、描画プリミティブに関する情報を受けた各処理回路において、たとえば入力情報に基づいて自モジュールの担当する領域に含まれるピクセルのみが描画されて対応するメモリモジュールに送られる。

【0031】

【発明の実施の形態】以下、本実施形態においては、パーソナルコンピュータなどに適用される、任意の3次元物体モデルに対する所望の3次元画像をCRT (Cathode Ray Tube) などのディスプレイ上に高速に表示する3次元コンピュータグラフィックスシステムについて説明する。

【0032】図3は、本発明に係る画像処理装置としての3次元コンピュータグラフィックスシステム10のシステム構成図である。

【0033】3次元コンピュータグラフィックスシステム10は、立体モデルを単位図形である三角形（ポリゴン）の組み合わせとして表現し、このポリゴンを描画することで表示画面の各ピクセルの色を決定し、ディスプレイに表示するポリゴンレンダリング処理を行うシステムである。また、3次元コンピュータグラフィックスシステム10では、平面上の位置を表現する（x, y）座標の他に、奥行きを表すz座標を用いて3次元物体を表し、この（x, y, z）の3つの座標で3次元空間の任意の一点を特定する。

【0034】図3に示すように、3次元コンピュータグラフィックスシステム10は、メインプロセッサ11、

メインメモリ12、I/Oインタフェース回路13、およびレンダリング回路14が、メインバス15を介して接続されている。以下、各構成要素の機能について説明する。

【0035】メインプロセッサ11は、たとえば、アプリケーションの進行状況などに応じて、メインメモリ12から必要なグラフィックデータを読み出し、このグラフィックデータに対して、座標変換、クリッピング(Cli  
pping)処理、ライティング(Lighting)処理などのジオメ  
トリ(Geometry)処理などを行い、ポリゴンレンダリング 10  
データを生成する。メインプロセッサ11は、ポリゴン  
レンダリングデータS11を、メインバス15を介して  
レンダリング回路14に出力する。

【0036】I/Oインタフェース回路13は、必要に  
応じて、外部から動きの制御情報またはポリゴンレンダ  
リングデータなどを入力し、これをメインバス15を介  
してレンダリング回路14に出力する。

【0037】レンダリング回路14に入力されるポリゴ  
ンレンダリングデータは、ポリゴンの各3頂点の(x,  
y, z, R, G, B, s, t, q)のデータを含んでい 20  
る。ここで、(x, y, z)データは、ポリゴンの頂点  
の3次元座標を示し、(R, G, B)データは、それぞ  
れ当該3次元座標における赤、緑、青の輝度値を示して  
いる。(s, t, q)データのうち、(s, t)は、対  
応するテクスチャの同次座標を示しており、qは同次項  
を示している。ここで、「s/q」および「t/q」  
に、それぞれテクスチャサイズUSIZEおよびVSI  
ZEを乗じて、実際のテクスチャ座標データ(u, v)  
が得られる。レンダリング回路14のグラフィックスメモ 30  
リ(具体的にはテクスチャバッファ)に記憶されたテク  
スチャデータへのアクセスは、テクスチャ座標データ  
(u, v)を用いて行われる。すなわち、ポリゴンレン  
ダリングデータは、三角形の各頂点の物理座標値と、そ  
れぞれの頂点の色とテクスチャデータである。

【0038】以下、レンダリング回路14について詳細  
に説明する。

【0039】図4に示すように、レンダリング回路14  
は、線形補間演算のための初期設定演算ブロックとしての  
DDA(Digital Differential Analyzer)セットアップ 40  
回路141、複数(本実施形態では4)の領域判定回  
路142-0~142-3、線形補間処理ブロックとし  
てのトライアングルDDA回路(以下、単にDDA回路  
という)143-0~143-3、テクスチャエンジン  
回路144-0~144-3、メモリインタフェース  
(I/F)回路145-0~145-3、専用メモリ  
I/F回路146、たとえばDRAMからなるグラフィッ  
クスメモリ147、CRTコントロール回路148、セ  
ットアップデータ用ブロードキャストバス149、およ  
びテクスチャバス150を有している。

【0040】本実施形態におけるレンダリング回路14 50

は、一つの半導体チップ内にロジック回路と少なくとも  
表示データとテクスチャデータとを記憶するグラフィッ  
クスメモリ147とが混載されている。

【0041】レンダリング回路14において、DDAセ  
ットアップ回路141の出力に対して複数個、本実施形  
態では4個の処理モジュール140-0~140-3が  
ブロードキャストバス149により並列に接続され、複  
数の処理モジュール140-0~140-3で処理デー  
タを共有し、各モジュール単位で割り当てられた担当描  
画領域を並列に処理する。

【0042】そして、グラフィックスメモリ147は、  
各処理モジュール140-0~140-3に対応して同  
一機能を有する複数(本実施形態では4個)のメモリモ  
ジュール147-0~147-3に分割されている。各  
処理モジュール140-0~140-3は、メモリモジ  
ュール147-0~147-3が処理の大きさ、たと  
えば4×4の矩形領域単位にインターリーブされており、  
メモリモジュール147-0と処理モジュール140-  
0、メモリモジュール147-1と処理モジュール14  
0-1、メモリモジュール147-2と処理モジュール  
140-2、およびメモリモジュール147-3と処理  
モジュール140-3は、担当領域が1対1に対応して  
おり、描画系について他のメモリモジュールに対するメ  
モリアクセスが発生しない。

【0043】図5および図6は、本実施形態に係る並列  
化処理の基本概念を説明するための図である。図5に示  
すように、本実施形態では、DDAセットアップ回路1  
41による同じ三角形プリミティブに関する情報(トラ  
イアングルデータ)をピクセル発生手段としての各処理  
モジュール140-0~140-3に送る。各処理モジ  
ュール140-0~140-3は、上述したように、各  
メモリモジュール147-0~147-3と対応付けら  
れており、自モジュールの担当する領域RGN0~RG  
N3に含まれるピクセルのみを描画して対応するメモリ  
モジュール147-0~147-3に送る。各処理モジ  
ュール140-0~140-3の担当領域RGN0~R  
GN3としては、所定の単位領域(本実施形態では16  
×16ピクセルの矩形領域)を処理モジュールの数だけ  
(本実施形態では4)分割した各領域がそれぞれ割り当  
てられる。なお、各処理モジュール140-0~140-  
3は、後述するように、複数の単位領域における担当  
領域をスキャンしながらピクセルを発生する。

【0044】たとえば図6(A)に示すような三角形を  
描画する場合、従来は矩形領域単位でピクセルを生成し  
ていたが、本実施形態では、図6(B)~(E)に示す  
ように、自モジュールの担当する領域RGN0~RGN  
3に含まれるピクセルのみを描画して対応するメモリモ  
ジュール147-0~147-3に送る。具体的には、  
図6(A)および図6(B)に示すように、領域RGN  
0に含まれるピクセルはメモリモジュール147-0に



送る。同様に、図6(A)および図6(C)に示すように、領域RGN1に含まれるピクセルはメモリモジュール147-1に送る。図6(A)および図6(D)に示すように、領域RGN2に含まれるピクセルはメモリモジュール147-2に送る。図6(A)および図6

(E)に示すように、領域RGN3に含まれるピクセルはメモリモジュール147-3に送る。

【0045】すなわち、本実施形態では、並列処理の粒度という観点からはピクセルレベルの並列処理を採用している。そして、ピクセルレベルの並列処理では並列度 10 に限界があるので、複数のプリミティブを同時に処理する。本実施形態では、同一のプリミティブに関して担当領域を複数の処理モジュール140-0~140-3に割り当てて並列に処理する。また、本実施形態では、各処理モジュール140-0~140-3とメモリモジュール147-0~147-3を直結できる処理分配とし、配線本数を削減し、小型化することができる。その結果、設計を容易とし、配線コスト、配線遅延を低減している。これにより、本実施形態では、三角形プリミティブデータはブロードキャスト、ピクセルは1対1の転送 20 となり、単純なデータ転送を実現している。また、たとえば巨大なトライアングルが入力された場合であっても、各処理モジュールで効率の良い並列処理が可能となっている。

【0046】一方、各処理モジュール140-0~140-3は、テクスチャリード系に関しては、他の処理モジュールに対するメモリアクセスを必要とする。この場合は、各処理モジュール140-0~140-3のテクスチャエンジン回路144-0~144-3は、図3に示すように、テクスチャバス150を介して専用メモリ 30 I/F回路146と通信してメモリアクセスを行う。

【0047】本実施形態において、処理モジュール140-0は、領域判定回路142-0、DDA回路143-0、テクスチャエンジン回路144-0、およびメモリI/F回路145-0を有している。そして、DDA回路143-0、テクスチャエンジン回路144-0、およびメモリI/F回路145-0により処理回路が構成される。

【0048】同様に、処理モジュール140-1は、領域判定回路142-1、DDA回路143-1、テクスチャエンジン回路144-1、およびメモリI/F回路145-1を有している。そして、DDA回路143-1、テクスチャエンジン回路144-1、およびメモリI/F回路145-1により処理回路が構成される。処理モジュール140- 40

3は、領域判定回路142-3、DDA回路143-3、テクスチャエンジン回路144-3、およびメモリI/F回路145-3を有している。そして、DDA回路143-3、テクスチャエンジン回路144-3、およびメモリI/F回路145-3により処理回路が構成される。

【0049】以下、レンダリング回路14の各ブロックの構成および機能について、図面に関連付けて順を追って説明する。

【0050】DDAセットアップ回路141は、後段の各処理モジュール140-0~140-3のDDA回路143-0~143-34において物理座標系上の三角形の各頂点の値を線形補間して、三角形の内部の各ピクセルの色と深さ情報を求めるに先立ち、メインバス15を介して入力したポリゴンレンダリングデータS11が示す(z, R, G, B, s, t, q)データについて、三角形の辺と水平方向の差分などを求めるセットアップ演算を行う。このセットアップ演算は、具体的には、開始点の値と終点の値と、開始点と終点との距離を用いて、単位長さ移動した場合における、求めようとしている値の変分を算出する。DDAセットアップ回路143は、算出した変分データを含む入力三角形に関する情報をブロードキャストバス149を通して各処理モジュール140-0~140-3の領域判定回路142-0~142-3に並列に出力する(分配する)。

【0051】DDAセットアップ回路141の機能について図7に関連付けてさらに説明する。上述したように、DDAセットアップ回路141の主な処理は、前段のジオメトリ処理を経て物理座標にまで落ちてきた各頂点における各種情報(色、テクスチャ座標)の与えられた三頂点P0(x0, y0)、P1(x1, y1)、P2(x2, y2)により構成される三角形内部で変分を求めて、後段の線形補間処理の基礎データを算出することである。三角形の描画はひとつひとつのピクセルの描画に集約されるが、そのために描画開始点における最初の値を求める必要がある。最初の描画点における各種情報は、頂点からその最初の描画点までの水平距離に水平方向の変分を掛けた値と、垂直距離に垂直方向の変分を掛けた値を足し合わせたものとなる。いったん目的の三角形の内部の一つの整数格子上の値が求まれば、対象の三角形内部のその他の格子点における値は変分の整数倍で求めることが可能となる。

【0052】なお、三角形の各頂点データは、たとえばx, y座標が16ビット、z座標が24ビット、RGBカラー値が各12ビット(=8+4)、s, t, qテクスチャ座標は各32ビット浮動少数値(IEEEフォーマット)等で構成される。

【0053】なお、このDDAセットアップ回路141は、たとえばDSP構造ではなく、ASIC手法により実装している。具体的には、図8に示すように、多段に



配置したレジスタ1411-1~1413間に並列に複数の演算ユニットを並列に配置した演算ユニット群1412-1~1412-3を挿入したフルデータパスロジック、換言すれば、同期パイプライン方式の時間並列構造として構成されている。

【0054】処理モジュール140-0の領域判定回路142-0は、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形に関する情報を受けて、たとえば三角形があらかじめ規定されている自モジュールの担当領域RGN0に含まれるか否かを判定し、含まれる場合には、入力情報を次段のDDA回路143-0に出力する。一方、領域判定回路142-0は、三角形があらかじめ規定されている自モジュールの担当領域RGN0に含まれていない場合には、入力情報を次段のDDA回路143-0に出力せずに棄てる。この領域判定回路142-0は、担当領域RGN0に三角形が含まれるか否かの判定を、入力情報に含まれるX方向およびY方向のそれぞれの開始点および終了点の座標を用いて行う。

【0055】処理モジュール140-0の担当領域RGN0は、たとえば1処理モジュール当たり1サイクルで4×4ピクセルの生成単位とすると、図9に示すように、2×2マトリクスの左上の領域である。したがって、領域判定回路142-0は、該当する三角形の一部がこの担当領域RGN0に掛かっている（含まれている）か否かの判定を行う。

【0056】図10および図11は、領域判定回路142の判定処理を概念的に説明するための図であって、図10は4並列の場合の例を示し、図11は8並列の場合の例を示している。

【0057】図10および図11の例では、三角形PM11およびPM12が処理モジュール140-0の担当領域RGN0に含まれる領域が全くないことから、領域判定回路142-0は、三角形があらかじめ規定されている自モジュールの担当領域RGN0に含まれていないものとして、入力情報を次段のDDA回路143-0に出力せずに棄てる。

【0058】なお、ここでは、処理モジュール140-0の領域判定回路142-0の処理を例に示しているが、他の処理モジュール140-1~140-3の領域判定回路142-1~142-3における処理も同様に行われる。

【0059】図12は、各処理モジュール140-0~140-3における領域判定回路142-0~142-3の判定処理動作を示すフローチャートである。図12に示すように、領域判定回路142（-0~3）は、まず、X方向において開始点座標から終了座標までの間に担当領域を含むか否かの判定を行う（ST1）。ステップST1において、担当領域を含んでいないと判定した場合には、領域判定回路142は、Y方向において開始

点座標から終了座標までの間に担当領域を含むか否かの判定を行う（ST2）。そして、ステップST2において、担当領域を含んでいないと判定した場合には、入力情報を次段のDDA回路143（-0~3）に出力せずに棄てる。一方、ステップST1またはステップST2において、担当領域を含んでいると判定した場合には、入力情報を次段のDDA回路143に出力する。

【0060】処理モジュール140-0のDDA回路143-0は、領域判定回路142-0から供給された変分データを用いて、三角形内部の担当領域の各ピクセルにおける線形補間された（z, R, G, B, s, t, q）データを算出する。DDA回路143-0は、各ピクセルの（x, y）データと、当該（x, y）座標における（z, R, G, B, s, t, q）データとを、DDAデータ（補間データ）としてテクスチャエンジン回路144-0に出力する。すなわち、DDA回路143-0は、パラメータデータに基づいて、たとえば三角形が自分が担当する領域である場合に、ラスタライゼーション（Rasterization）を行う。具体的には、その三角形が自分が担当する領域の場合に、各種データ（z、テクスチャ座標、カラーなど）をラスタライズする。この場合、生成単位は、1ローカルモジュール当たり1サイクルで4×4ピクセルである。

【0061】DDA回路143-0の機能について図13に関連付けてさらに説明する。上述したように、前段のDDAセットアップ回路141により、三角形の描画開始点における最初の値、水平方向（X方向）および垂直方向（Y方向）における先出の各種情報の傾き情報が準備される。ここでの基本的処理は、与えられた三角形の内部に含まれる整数格子点の値を求めることであり、その処理の実体は描画開始点からの整数距離と傾きの掛け算ということになる。また、実際には掛け算を行うというより、水平方向（X方向）に1ピクセル進め水平方向の傾き分を足しこめば1ピクセル進んだ分の値が求められるということになる。よって、計算内容としては、一定値の加算処理となる。

【0062】ただし、本実施形態に係るDDA回路は、図1に示すように一列のピクセル生成処理を行う場合とスキャンの仕方が異なる。たとえば図13（A）に示すような三角形を描画する場合、従来は矩形領域単位でピクセルを生成していた。すなわち、図中、破線で示す処理開始点からX方向にスキャンしていき、三角形の外に出たらY方向に移動して再びスキャンを行っていた。これに対して、本実施形態では、処理開始点から自モジュールの担当領域に移動する。そして、X方向にスキャンする。三角形の外に出たらY方向に移動して再びX方向にスキャンする。すなわち、担当領域を割り振って並列処理を行う本実施形態では、スキャン自の移動の幅が、従来と異なる。処理モジュール140-0のDDA回路143-0は、上述したように、2×2分割の矩形

領域の左上領域が担当領域である。本例では、図13(B)に示すように、破線で示す処理開始点が自モジュールの担当領域RGN0であることから、DDA回路143-0は、次に、Y方向(図面で上方向)に移動し、再びX方向(図面で右方向)にスキャンする。

【0063】なお、ここでは、処理モジュール140-0のDDA回路143-0の処理を例に示しているが、他の処理モジュール140-1~140-3のDDA回路143-1~143-3における処理も同様に行われる。

【0064】図14は、各処理モジュール140-0~140-3におけるDDA回路143-0~143-3の判定処理動作を示すフローチャートである。図14に示すように、DDA回路143(-0~3)は、まず、処理開始点からX方向およびY方向それぞれについて終了方向に最も近い担当領域へ移動する(ST11)。次に、担当領域の中で、三角形(トライアングル)の内部にある部分についてピクセルを生成する(ST12)。

【0065】テクスチャエンジン回路144-0は、「s/q」および「t/q」の算出処理、テクスチャ座標データ(u, v)の算出処理、グラフィックスメモリ147に含まれるテクスチャバッファからの(R, G, B)データの読み出し処理等をパイプライン方式で行う。なお、テクスチャエンジン回路144-0は、他の処理モジュール140-1~140-3のテクスチャエンジン回路144-1~144-3と協働して、たとえば所定の矩形内に位置する8ピクセルについての処理を同時に並行して行う。テクスチャエンジン回路144-0は、自モジュールのDDA回路144-0によるDDAデータが示す(s, t, q)データについて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とを行う。また、テクスチャエンジン回路144-0は、除算結果である「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEを乗じて、テクスチャ座標データ(u, v)を生成する。

【0066】また、テクスチャエンジン回路144-0は、テクスチャバス150、専用メモリI/F回路146を介して、グラフィックスメモリ147に、生成したテクスチャ座標データ(u, v)を含む読み出し要求を出力し、専用メモリI/F回路146を介して、グラフィックスメモリ147に含まれるテクスチャバッファに記憶されているテクスチャデータを読み出すことで、(s, t)データに対応したテクスチャアドレスに記憶された(R, G, B)データを得る。テクスチャエンジン回路144-0は、読み出した(R, G, B)データと、前段のDDA回路143-0からのDDAデータに含まれる(R, G, B)データとを、それぞれ掛け合わせるなどして、ピクセルデータを生成する。テクスチャエンジン回路144-0は、このピクセルデータを最終

的にピクセルのカラー値としてメモリI/F回路145-0に出力する。この場合、テクスチャエンジン回路144-0は、読み出されたテクスチャデータと、(u, v)アドレスは算出時に得た小数部を使って4近傍補間などのフィルタリング処理を行う。

【0067】なお、テクスチャバッファには、MIPMAP(複数解像度テクスチャ)などの複数の縮小率に対応したテクスチャデータが記憶されている。ここで、何れの縮小率のテクスチャデータを用いるかは、所定のアルゴリズムを用いて、前記三角形単位で決定される。

【0068】テクスチャエンジン回路144-0は、フルカラー方式の場合には、テクスチャバッファから読み出した(R, G, B)データを直接用いる。一方、テクスチャエンジン回路144-0は、インデックスカラー方式の場合には、あらかじめ作成しておいたカラーインデックステーブルのデータを、テクスチャカラーlookupアップテーブル(CLUT)バッファより内蔵するSRAM等で構成した一時保管バッファに転送しておいて、このカラーlookupアップテーブルを用いて、テクスチャバッファから読み出したカラーインデックスに対応する(R, G, B)データを得る。たとえばカラーlookupアップテーブルがSRAMで構成された場合、カラーインデックスをSRAMのアドレスに入力すると、その出力には実際の(R, G, B)データが出てくるといった使い方となる。

【0069】メモリI/F回路145-0は、テクスチャエンジン回路144-0から入力したピクセルデータに対応するzデータと、メモリモジュール147-0に含まれるzバッファに記憶されているzデータとの比較を行い、入力したピクセルデータによって描画される画像が、前回、メモリモジュール147-0(ディスプレイバッファ)に書き込まれた画像より、手前(視点側)に位置するか否かを判定し、手前に位置する場合には、画像データに対応するzデータでzバッファに記憶されたzデータを更新する。また、メモリI/F回路145-0は、(R, G, B)データをメモリモジュール147-0に書き込む。メモリI/F回路145-0は、今から描画しようとしているピクセルアドレスに対応するピクセルデータを格納しているメモリモジュール147-0に対して、該当アドレスからピクセルデータをモディファイ書き込みするために読み出し、モディファイ後同じアドレスへ書き戻す。隠れ面処理を行なう場合には、やはり同じように今から描画しようとしているピクセルアドレスに対応する奥行きデータを格納しているメモリブロックに対して、該当アドレスから奥行きデータをモディファイ書き込みするため読み出し、必要ならばモディファイ後同じアドレスへ書き戻す。また、メモリI/F回路145-0は、テクスチャエンジン回路144-0により供給されたピクセルレベルの処理結果を受けて、ピクセルレベルの処理における各種テストをパス

したピクセルデータをメモリモジュール147-0に描画する。

【0070】処理モジュール140-1の領域判定回路142-1は、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形に関する情報を受けて、たとえば三角形があらかじめ規定されている自モジュールの担当領域RGN1に含まれるか否かを判定し、含まれる場合には、入力情報を次段のDDA回路143-1に出力する。一方、領域判定回路142-1は、三角形があらかじめ規定されている自モジュールの担当領域RGN1に含まれていない場合には、入力情報を次段のDDA回路143-1に出力せずに棄てる。この領域判定回路142-1は、担当領域RGN1に三角形が含まれるか否かの判定を、入力情報に含まれるX方向およびY方向のそれぞれの開始点および終了点の座標を用いて行う。

【0071】処理モジュール140-1の担当領域RGN1は、たとえば1処理モジュール当たり1サイクルで2×2ピクセルの生成単位とすると、図9に示すように、2×2マトリックスの右上の領域である。したがって、領域判定回路142-1は、該当する三角形の一部がこの担当領域RGN1に掛かっている（含まれている）か否かの判定を行う。

【0072】たとえば図10の例では、三角形PM11が処理モジュール140-1の担当領域RGN1に含まれる領域が全くないことから、領域判定回路142-1は、三角形PM11があらかじめ規定されている自モジュールの担当領域RGN1に含まれていないものとして、入力情報を次段のDDA回路143-1に出力せずに棄てる。一方、図10の例における三角形PM12は、処理モジュール140-1の担当領域RGN1に含まれる領域があることから、領域判定回路142-1は、三角形PM12があらかじめ規定されている自モジュールの担当領域RGN1に含まれているものとして、入力情報を次段のDDA回路143-1に出力する。

【0073】処理モジュール140-1のDDA回路143-1は、領域判定回路142-1から供給された変分データを用いて、三角形内部の担当領域の各ピクセルにおける線形補間された(z, R, G, B, s, t, q)データを算出する。DDA回路143-1は、各ピクセルの(x, y)データと、当該(x, y)座標における(z, R, G, B, s, t, q)データとを、DDAデータ(補間データ)としてテクスチャエンジン回路144-1に出力する。すなわち、DDA回路143-1は、パラメータデータに基づいて、たとえば三角形が自分が担当する領域である場合に、ラスタライゼーション(Rasterization)を行う。具体的には、その三角形が自分が担当する領域の場合に、各種データ(z, テクスチャ座標、カラーなど)をラスタライズする。この場合、生成単位は、1ローカルモジュール

当たり1サイクルで4×4ピクセルである。

【0074】DDA回路143-1は、前段のDDAセットアップ回路141により、三角形の描画開始点における最初の値、水平方向(X方向)および垂直方向(Y方向)における先出の各種情報の傾き情報が準備される。ここでの基本的処理は、与えられた三角形の内部に含まれる整数格子上の値を求めることであり、その処理の実体は描画開始点からの整数距離と傾きの掛け算ということになる。また、実際には掛け算を行うというより、水平方向(X方向)に1ピクセル進め水平方向の傾き分を足しこめば1ピクセル進んだ分の値が求められるということになる。よって、計算内容としては、一定値の加算処理となる。

【0075】但し、本実施形態に係るDDA回路は、図1に示すように一列のピクセル生成処理を行う場合とスキヤンの仕方が異なる。たとえば図13(A)に示すような三角形を描画する場合、従来は矩形領域単位でピクセルを生成していた。すなわち、図中、破線で示す処理開始点からX方向にスキヤンしていき、三角形の外に出たらY方向に移動して再びスキヤンを行っていた。これに対して、本実施形態では、処理開始点から自モジュールの担当領域に移動する。そして、X方向にスキヤンする。三角形の外に出たらY方向に移動して再びX方向にスキヤンする。すなわち、担当領域を割り振って並列処理を行う本実施形態では、スキヤン自の移動の幅が、従来と異なる。処理モジュール140-1のDDA回路143-1は、上述したように、2×2分割の矩形領域の右上領域が担当領域である。本例では、図13(C)に示すように、破線で示す処理開始点からX方向にスキヤンして自モジュールの担当領域RGN1に移動する。そして、DDA回路143-1は、次に、Y方向(図面で上方向)に移動し、再びX方向(図面で右方向)にスキヤンする。

【0076】テクスチャエンジン回路144-1は、「s/q」および「t/q」の算出処理、テクスチャ座標データ(u, v)の算出処理、グラフィックスメモリ147に含まれるテクスチャバッファからの(R, G, B)データの読み出し処理等をパイプライン方式で行う。なお、テクスチャエンジン回路144-1は、他の処理モジュール140-0、140-2~140-3のテクスチャエンジン回路144-0、144-2~144-3と協働して、たとえば所定の矩形内に位置する8ピクセルについての処理を同時に並行して行う。テクスチャエンジン回路144-1は、自モジュールのDDA回路144-1によるDDAデータが示す(s, t, q)データについて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とを行う。また、テクスチャエンジン回路144-1は、除算結果である「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEを乗じて、テクス

チャ座標データ (u, v) を生成する。

【0077】また、テクスチャエンジン回路144-1は、テクスチャバス150、専用メモリI/F回路146を介して、グラフィックスメモリ147に、生成したテクスチャ座標データ (u, v) を含む読み出し要求を出力し、専用メモリI/F回路146を介して、グラフィックスメモリ147に含まれるテクスチャバッファに記憶されているテクスチャデータを読み出すことで、

(s, t) データに対応したテクスチャアドレスに記憶された (R, G, B) データを得る。テクスチャエンジン回路144-1は、読み出した (R, G, B) データと、前段のDDA回路143-1からのDDAデータに含まれる (R, G, B) データとを、それぞれ掛け合わせるなどして、ピクセルデータを生成する。テクスチャエンジン回路144-1は、このピクセルデータを最終的にピクセルのカラー値としてメモリI/F回路145-1に出力する。この場合、テクスチャエンジン回路144-1は、読み出されたテクスチャデータと、(u, v) アドレスは算出時に得た小数部を使って4近傍補間などのフィルタリング処理を行う。

【0078】なお、テクスチャバッファには、MIPMAP (複数解像度テクスチャ) などの複数の縮小率に対応したテクスチャデータが記憶されている。ここで、何れの縮小率のテクスチャデータを用いるかは、所定のアルゴリズムを用いて、前記三角形単位で決定される。

【0079】テクスチャエンジン回路144-1は、フルカラー方式の場合には、テクスチャバッファから読み出した (R, G, B) データを直接用いる。一方、テクスチャエンジン回路144-1は、インデックスカラー方式の場合には、あらかじめ作成しておいたカラーインデックステーブルのデータを、テクスチャカラーlookupアップテーブル (CLUT) バッファより内蔵するSRAM等で構成した一時保管バッファに転送しておいて、このカラーlookupアップテーブルを用いて、テクスチャバッファから読み出したカラーインデックスに対応する (R, G, B) データを得る。たとえばカラーlookupアップテーブルがSRAMで構成された場合、カラーインデックスをSRAMのアドレスに入力すると、その出力には実際の (R, G, B) データが出てくるといった使い方となる。

【0080】メモリI/F回路145-1は、テクスチャエンジン回路144-1から入力したピクセルデータに対応するzデータと、メモリモジュール147-1に含まれるzバッファに記憶されているzデータとの比較を行い、入力したピクセルデータによって描画される画像が、前回、メモリモジュール147-1 (ディスプレイバッファ) に書き込まれた画像より、手前 (視点側) に位置するか否かを判定し、手前に位置する場合には、画像データに対応するzデータでzバッファに記憶されたzデータを更新する。また、メモリI/F回路145

-1は、(R, G, B) データをメモリモジュール147-1に書き込む。メモリI/F回路145-1は、今から描画しようとしているピクセルアドレスに対応するピクセルデータを格納しているメモリモジュール147-1に対して、該当アドレスからピクセルデータをモディファイ書き込みするために読み出し、モディファイ後同じアドレスへ書き戻す。隠れ面処理を行なう場合には、やはり同じように今から描画しようとしているピクセルアドレスに対応する奥行きデータを格納しているメモリブロックに対して、該当アドレスから奥行きデータをモディファイ書き込みするため読み出し、必要ならばモディファイ後同じアドレスへ書き戻す。また、メモリI/F回路145-1は、テクスチャエンジン回路144-1により供給されたピクセルレベルの処理結果を受けて、ピクセルレベルの処理における各種テストをパスしたピクセルデータをメモリモジュール147-1に描画する。

【0081】処理モジュール140-2の領域判定回路142-2は、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形に関する情報を受けて、たとえば三角形があらかじめ規定されている自モジュールの担当領域RGN2に含まれるか否かを判定し、含まれる場合には、入力情報を次段のDDA回路143-2に出力する。一方、領域判定回路142-2は、三角形があらかじめ規定されている自モジュールの担当領域RGN2に含まれていない場合には、入力情報を次段のDDA回路143-2に出力せずに棄てる。この領域判定回路142-2は、担当領域RGN2に三角形が含まれるか否かの判定を、入力情報に含まれるX方向およびY方向のそれぞれの開始点および終了点の座標を用いて行う。

【0082】処理モジュール140-2の担当領域RGN2は、たとえば1処理モジュール当たり1サイクルで4×4ピクセルの生成単位とすると、図9に示すように、2×2分割の左下の領域である。したがって、領域判定回路142-2は、該当する三角形の一部がこの担当領域RGN2に掛かっている (含まれている) か否かの判定を行う。

【0083】たとえば図10の例では、三角形PM11が処理モジュール140-2の担当領域RGN2に含まれる領域があることから、領域判定回路142-2は、三角形PM11があらかじめ規定されている自モジュールの担当領域RGN2に含まれているものとして、入力情報を次段のDDA回路143-2に出力する。一方、図10の例における三角形PM12は、処理モジュール140-2の担当領域RGN2に含まれる領域が全くないことから、領域判定回路142-2は、三角形PM12があらかじめ規定されている自モジュールの担当領域RGN2に含まれていないものとして、入力情報を次段のDDA回路143-2に出力せずに棄てる。

【0084】処理モジュール140-2のDDA回路143-2は、領域判定回路142-2から供給された変分データを用いて、三角形内部の担当領域の各ピクセルにおける線形補間された(z, R, G, B, s, t, q)データを算出する。DDA回路143-2は、各ピクセルの(x, y)データと、当該(x, y)座標における(z, R, G, B, s, t, q)データとを、DDAデータ(補間データ)としてテクスチャエンジン回路144-2に出力する。すなわち、DDA回路143-2は、パラメータデータに基づいて、たとえば三角形が自分が担当する領域である場合に、ラスタライゼーション(Rasterization)を行う。具体的には、その三角形が自分が担当する領域の場合に、各種データ(z, テクスチャ座標、カラーなど)をラスタライズする。この場合、生成単位は、1ローカルモジュール当たり1サイクルで4×4ピクセルである。

【0085】DDA回路143-2は、前段のDDAセットアップ回路141により、三角形の描画開始点における最初の値、水平方向(X方向)および垂直方向(Y方向)における先出の各種情報の傾き情報が準備される。ここでの基本的処理は、与えられた三角形の内部に含まれる整数格子上の値を求めることであり、その処理の実体は描画開始点からの整数距離と傾きの掛け算ということになる。また、実際には掛け算を行うというより、水平方向(X方向)に1ピクセル進め水平方向の傾き分を足しこめば1ピクセル進んだ分の値が求められるということになる。よって、計算内容としては、一定値の加算処理となる。

【0086】但し、本実施形態に係るDDA回路は、図1に示すように一列のピクセル生成処理を行う場合とスキヤンの仕方が異なる。たとえば図13(A)に示すような三角形を描画する場合、従来は矩形領域単位でピクセルを生成していた。すなわち、図中、破線で示す処理開始点からX方向にスキヤンしていき、三角形の外に出たらY方向に移動して再びスキヤンを行っていた。これに対して、本実施形態では、処理開始点から自モジュールの担当領域に移動する。そして、X方向にスキヤンする。三角形の外に出たらY方向に移動して再びX方向にスキヤンする。すなわち、担当領域を割り振って並列処理を行う本実施形態では、スキヤン自の移動の幅が、従来と異なる。処理モジュール140-2のDDA回路143-2は、上述したように、2×2の矩形領域の左下領域が担当領域である。本例では、図13(D)に示すように、破線で示す処理開始点からY方向(図面で上方向)にスキヤンして自モジュールの担当領域RGN2に移動する。そして、DDA回路143-2は、次に、X方向(図面で右方向)にスキヤンする。

【0087】テクスチャエンジン回路144-2は、「s/q」および「t/q」の算出処理、テクスチャ座標データ(u, v)の算出処理、グラフィックスメモリ

147に含まれるテクスチャバッファからの(R, G, B)データの読み出し処理等をパイプライン方式で行う。なお、テクスチャエンジン回路144-2は、他の処理モジュール140-0、140-1、140-3のテクスチャエンジン回路144-0、144-1、144-3と協働して、たとえば所定の矩形内に位置する8ピクセルについての処理を同時に並行して行う。テクスチャエンジン回路144-2は、自モジュールのDDA回路144-2によるDDAデータが示す(s, t, q)データについて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とを行う。また、テクスチャエンジン回路144-2は、除算結果である「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEを乗じて、テクスチャ座標データ(u, v)を生成する。

【0088】また、テクスチャエンジン回路144-2は、テクスチャバス150、専用メモリI/F回路146を介して、グラフィックスメモリ147に、生成したテクスチャ座標データ(u, v)を含む読み出し要求を出力し、専用メモリI/F回路146を介して、グラフィックスメモリ147に含まれるテクスチャバッファに記憶されているテクスチャデータを読み出すことで、

(s, t)データに対応したテクスチャアドレスに記憶された(R, G, B)データを得る。テクスチャエンジン回路144-2は、読み出した(R, G, B)データと、前段のDDA回路143-2からのDDAデータに含まれる(R, G, B)データとを、それぞれ掛け合わせるなどして、ピクセルデータを生成する。テクスチャエンジン回路144-2は、このピクセルデータを最終的にピクセルのカラー値としてメモリI/F回路145-2に出力する。この場合、テクスチャエンジン回路144-2は、読み出されたテクスチャデータと、(u, v)アドレスは算出時に得た小数部を使って4近傍補間などのフィルタリング処理を行う。

【0089】なお、テクスチャバッファには、MIPMAP(複数解像度テクスチャ)などの複数の縮小率に対応したテクスチャデータが記憶されている。ここで、何れの縮小率のテクスチャデータを用いるかは、所定のアルゴリズムを用いて、前記三角形単位で決定される。

【0090】テクスチャエンジン回路144-2は、フルカラー方式の場合には、テクスチャバッファから読み出した(R, G, B)データを直接用いる。一方、テクスチャエンジン回路144-2は、インデックスカラー方式の場合には、あらかじめ作成しておいたカラーインデックステーブルのデータを、テクスチャカラーlookupアップテーブル(CLUT)バッファより内蔵するSRAM等で構成した一時保管バッファに転送しておいて、このカラーlookupアップテーブルを用いて、テクスチャバッファから読み出したカラーインデックスに対応する(R, G, B)データを得る。たとえばカラーlookupア

ップテーブルがSRAMで構成された場合、カラーインデックスをSRAMのアドレスに入力すると、その出力には実際の(R, G, B)データが出てくるといった使い方となる。

【0091】メモリI/F回路145-2は、テクスチャエンジン回路144-2から入力したピクセルデータに対応するzデータと、メモリモジュール147-2に含まれるzバッファに記憶されているzデータとの比較を行い、入力したピクセルデータによって描画される画像が、前回、メモリモジュール147-2(ディスプレイバッファ)に書き込まれた画像より、手前(視点側)に位置するか否かを判定し、手前に位置する場合には、画像データに対応するzデータでzバッファに記憶されたzデータを更新する。また、メモリI/F回路145-2は、(R, G, B)データをメモリモジュール147-1に書き込む。メモリI/F回路145-2は、今から描画しようとしているピクセルアドレスに対応するピクセルデータを格納しているメモリモジュール147-2に対して、該当アドレスからピクセルデータをモディファイ書き込みするために読み出し、モディファイ後

同じアドレスへ書き戻す。隠れ面処理を行なう場合には、やはり同じように今から描画しようとしているピクセルアドレスに対応する奥行きデータを格納しているメモリブロックに対して、該当アドレスから奥行きデータをモディファイ書き込みするため読み出し、必要ならばモディファイ後同じアドレスへ書き戻す。また、メモリI/F回路145-2は、テクスチャエンジン回路144-2により供給されたピクセルレベルの処理結果を受けて、ピクセルレベルの処理における各種テストをパスしたピクセルデータをメモリモジュール147-2に描

画する。

【0092】処理モジュール140-3の領域判定回路142-3は、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形に関する情報を受けて、たとえば三角形があらかじめ規定されている自モジュールの担当領域RGN3に含まれるか否かを判定し、含まれる場合には、入力情報を次段のDDA回路143-3に出力する。一方、領域判定回路142-3は、三角形があらかじめ規定されている自モジュールの担当領域RGN3に含まれていない場合には、入力情報を次段のDDA回路143-3に出力せずに棄てる。この領域判定回路142-3は、担当領域RGN1に三角形が含まれるか否かの判定を、入力情報に含まれるX方向およびY方向のそれぞれの開始点および終了点の座標を用いて行う。

【0093】処理モジュール140-3の担当領域RGN3は、たとえば1処理モジュール当たり1サイクルで4×4ピクセルの生成単位とすると、図9に示すように、2×2分割の右下の領域である。したがって、領域判定回路142-3は、該当する三角形の一部がこの担

当領域RGN3に掛かっている(含まれている)か否かの判定を行う。

【0094】たとえば図10の例では、三角形PM11およびPM12は、処理モジュール140-3の担当領域RGN3に含まれる領域があることから、領域判定回路142-3は、三角形PM11およびPM12があらかじめ規定されている自モジュールの担当領域RGN3に含まれているものとして、入力情報を次段のDDA回路143-3に出力する。

【0095】処理モジュール140-3のDDA回路143-3は、領域判定回路142-3から供給された変分データを用いて、三角形内部の担当領域の各ピクセルにおける線形補間された(z, R, G, B, s, t, q)データを算出する。DDA回路143-3は、各ピクセルの(x, y)データと、当該(x, y)座標における(z, R, G, B, s, t, q)データとを、DDAデータ(補間データ)としてテクスチャエンジン回路144-3に出力する。すなわち、DDA回路143-3は、パラメータデータに基づいて、たとえば三角形が自分が担当する領域である場合に、ラスタライゼーション(Rasterization)を行う。具体的には、その三角形が自分が担当する領域の場合に、各種データ(z, テクスチャ座標、カラーなど)をラスタライズする。この場合、生成単位は、1ローカルモジュール当たり1サイクルで4×4ピクセルである。

【0096】DDA回路143-3は、前段のDDAセットアップ回路141により、三角形の描画開始点における最初の値、水平方向(X方向)および垂直方向(Y方向)における先出の各種情報の傾き情報が準備される。ここでの基本的処理は、与えられた三角形の内部に含まれる整数格子上の値を求めることであり、その処理の実体は描画開始点からの整数距離と傾きの掛け算ということになる。また、実際には掛け算を行うというより、水平方向(X方向)に1ピクセル進め水平方向の傾き分を足しこめば1ピクセル進んだ分の値が求められるということになる。よって、計算内容としては、一定値の加算処理となる。

【0097】但し、本実施形態に係るDDA回路は、図1に示すように一列のピクセル生成処理を行う場合とスキヤンの仕方が異なる。たとえば図13(A)に示すような三角形を描画する場合、従来は矩形領域単位でピクセルを生成していた。すなわち、図中、破線で示す処理開始点からX方向にスキヤンしていき、三角形の外に出たらY方向に移動して再びスキヤンを行っていた。これに対して、本実施形態では、処理開始点から自モジュールの担当領域に移動する。そして、X方向にスキヤンする。三角形の外に出たらY方向に移動して再びX方向にスキヤンする。すなわち、担当領域を割り振って並列処理を行う本実施形態では、スキヤン自の移動の幅が、従来と異なる。処理モジュール140-3のDDA回路



143-3は、上述したように、2×2分割の矩形領域の右下領域が担当領域である。本例では、図13(E)に示すように、破線で示す処理開始点からX方向にスキャンしてさらにY方向にスキャンして自モジュールの担当領域RGN3に移動する。そして、DDA回路143-3は、次に、X方向(図面で右方向)にスキャンする。

【0098】テキストエンジン回路144-3は、「s/q」および「t/q」の算出処理、テキスト座標データ(u, v)の算出処理、グラフィックスメモリ147に含まれるテキストバッファからの(R, G, B)データの読み出し処理等をパイプライン方式で行う。なお、テキストエンジン回路144-3は、他の処理モジュール140-0~140-2のテキストエンジン回路144-0~144-2と協働して、たとえば所定の矩形内に位置する8ピクセルについての処理を同時に並行して行う。テキストエンジン回路144-3は、自モジュールのDDA回路144-3によるDDAデータが示す(s, t, q)データについて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とを行う。また、テキストエンジン回路144-3は、除算結果である「s/q」および「t/q」に、それぞれテキストサイズUSIZEおよびVSIZEを乗じて、テキスト座標データ(u, v)を生成する。

【0099】また、テキストエンジン回路144-3は、テキストバス150、専用メモリI/F回路146を介して、グラフィックスメモリ147に、生成したテキスト座標データ(u, v)を含む読み出し要求を出力し、専用メモリI/F回路146を介して、グラフィックスメモリ147に含まれるテキストバッファに記憶されているテキストデータを読み出すことで、

(s, t)データに対応したテキストアドレスに記憶された(R, G, B)データを得る。テキストエンジン回路144-3は、読み出した(R, G, B)データと、前段のDDA回路143-3からのDDAデータに含まれる(R, G, B)データとを、それぞれ掛け合わせるなどして、ピクセルデータを生成する。テキストエンジン回路144-3は、このピクセルデータを最終的にピクセルのカラー値としてメモリI/F回路145-3に出力する。この場合、テキストエンジン回路144-3は、読み出されたテキストデータと、(u, v)アドレスは算出時に得た小数部を使って4近傍補間などのフィルタリング処理を行う。

【0100】なお、テキストバッファには、MIPMAP(複数解像度テキスト)などの複数の縮小率に対応したテキストデータが記憶されている。ここで、何れの縮小率のテキストデータを用いるかは、所定のアルゴリズムを用いて、前記三角形単位で決定される。

【0101】テキストエンジン回路144-3は、フ

ルカラー方式の場合には、テキストバッファから読み出した(R, G, B)データを直接用いる。一方、テキストエンジン回路144-3は、インデックスカラー方式の場合には、あらかじめ作成しておいたカラーインデックステーブルのデータを、テキストカラーlookupアップテーブル(CLUT)バッファより内蔵するSRAM等で構成した一時保管バッファに転送しておいて、このカラーlookupアップテーブルを用いて、テキストバッファから読み出したカラーインデックスに対応する(R, G, B)データを得る。たとえばカラーlookupアップテーブルがSRAMで構成された場合、カラーインデックスをSRAMのアドレスに入力すると、その出力には実際の(R, G, B)データが出てくるといった使い方となる。

【0102】メモリI/F回路145-3は、テキストエンジン回路144-3から入力したピクセルデータに対応するzデータと、メモリモジュール147-3に含まれるzバッファに記憶されているzデータとの比較を行い、入力したピクセルデータによって描画される画像が、前回、メモリモジュール147-3(ディスプレイバッファ)に書き込まれた画像より、手前(視点側)に位置するか否かを判定し、手前に位置する場合には、画像データに対応するzデータでzバッファに記憶されたzデータを更新する。また、メモリI/F回路145-3は、(R, G, B)データをメモリモジュール147-3に書き込む。メモリI/F回路145-3は、今から描画しようとしているピクセルアドレスに対応するピクセルデータを格納しているメモリモジュール147-3に対して、該当アドレスからピクセルデータをモディファイ書き込みするために読み出し、モディファイ後同じアドレスへ書き戻す。隠れ面処理を行なう場合には、やはり同じように今から描画しようとしているピクセルアドレスに対応する奥行きデータを格納しているメモリブロックに対して、該当アドレスから奥行きデータをモディファイ書き込みするため読み出し、必要ならばモディファイ後同じアドレスへ書き戻す。また、メモリI/F回路145-3は、テキストエンジン回路144-3により供給されたピクセルレベルの処理結果を受けて、ピクセルレベルの処理における各種テストをパスしたピクセルデータをメモリモジュール147-3に描画する。

【0103】専用メモリI/F回路146は、今から描画しようとしているピクセルにおけるテキストアドレスに対応したテキストデータを格納しているメモリブロックをそのテキストアドレスより算出し、グラフィックスメモリ147の所定のメモリモジュールにのみ読み出し要求を出すことにより、テキストデータを読み出す。また、専用メモリI/F回路146は、テキストエンジン回路144-0~144-3から生成されたテキスト座標データ(u, v)を含む読み出し要求を

10

20

30

40

50



受けた場合には、SRAM等のバッファに記憶された(R, G, B)データを読み出す。また、専用メモリI/F回路146は、CRTコントロール回路148から表示データを読み出す要求を受けた場合には、この要求に応じて、グラフィックスメモリ147(具体的にはディスプレイバッファ)から一定の固まり、たとえば8ピクセルあるいは16ピクセル単位で表示データを読み出す。

【0104】グラフィックスメモリ147は、たとえばテクスチャバッファ、ディスプレイバッファ、zバッファおよびテクスチャCLUT(Color Look Up Table)バッファとして機能する。また、グラフィックスメモリ147は、前述したように、同一機能を有する複数(本実施形態では4個)のメモリモジュール147-0~147-3に分割されている。また、グラフィックスメモリ147には、より多くのテクスチャデータを格納するために、インデックスカラーにおけるインデックスと、そのためのカラーlookupテーブル値が、テクスチャCLUTバッファに格納されている。インデックスおよびカラーlookupテーブル値は、上述したように、テクスチャ処理に使われる。すなわち、通常はR, G, Bそれぞれ8ビットの合計24ビットでテクスチャ要素を表現するが、それではデータ量が膨らむため、あらかじめ選んでおいたたとえば256色等の中から一つの色を選んで、そのデータをテクスチャ処理に使う。このことで256色であればそれぞれのテクスチャ要素は8ビットで表現できることになる。インデックスから実際のカラーへの変換テーブルは必要になるが、テクスチャの解像度が高くなるほど、よりコンパクトなテクスチャデータとすることが可能となる。これにより、テクスチャデータの圧縮が可能となり、内蔵メモリの効率良い利用が可能となる。

【0105】以上のように、DDAセットアップ回路141、DDA回路143-0~143-3、テクスチャエンジン回路144-0~144-3、メモリI/F回路145-0~145-3等における所定を経て、最終的なメモリアクセスがピクセル(Pixel; Picture Cell Element)という描画ピクセル単位になる。

【0106】CRTコントロール回路148は、与えられた水平および垂直同期信号に同期して、図示しないCRTに表示する表示アドレスを発生し、グラフィックスメモリ147に含まれるディスプレイバッファから表示データを読み出す要求を専用メモリI/F回路146に出力する。この要求に応じて、専用メモリI/F回路146は、グラフィックスメモリ147(ディスプレイバッファ)から一定の固まりで表示データを読み出す。CRTコントロール回路148は、グラフィックスメモリ147から読み出した表示データを記憶するたとえばFIFO回路を内蔵し、一定の時間間隔で、RGBのインデックス値を発生する。CRTコントロール回路148

は、各インデックス値に対応するR, G, Bデータを記憶しており、発生したRGBのインデックス値に対応するデジタル形式のR, G, Bデータを、図示しないD/Aコンバータ(Digital/Analog Converter)に転送し、アナログ形式のR, G, Bデータを生成する。CRTコントロール回路148は、この生成されたR, G, Bデータを図示しないCRTに出力する。

【0107】次に、上記構成による動作を、図15~図20に関連付けて説明する。なおここでは、図15に示すように、3つの三角形プリミティブPM101, PM102, PM103を順次入力して描画処理する場合を説明する。この例では、プリミティブPM101は、全処理モジュール140-0~140-3の担当領域RGN0~RGN3に掛かっている(含まれている)。プリミティブPM102は、処理モジュール140-0~140-1の担当領域RGN0, RGN1に掛かっている(含まれている)が、処理モジュール140-2, 140-3の担当領域RGN2, RGN3には掛かっていない(含まれていない)。プリミティブPM103は、全処理モジュール140-0~140-3の担当領域RGN0~RGN3に掛かっている(含まれている)。

【0108】3次元コンピュータグラフィックスシステム10においては、グラフィックス描画等のデータは、メインプロセッサ11のメインメモリ12、あるいは外部からのグラフィックスデータを受けるI/Oインタフェース回路13からメインバス15を介してレンダリング回路14に与えられる。なお、必要に応じて、グラフィックス描画等のデータは、メインプロセッサ11等において、座標変換、クリップ処理、ライティング処理等のジオメトリ処理が行われる。ジオメトリ処理が終わったグラフィックスデータは、三角形の各3頂点の頂点座標x, y, z、輝度値R, G, B、描画しようとしているピクセルと対応するテクスチャ座標s, t, qとからなるポリゴンレンダリングデータS11となる。

【0109】本例では、図16に示すように、3つの三角形プリミティブPM101, PM102, PM103に係るポリゴンレンダリングデータS11が、レンダリング回路14のDDAセットアップ回路141に順々に転送される。

【0110】DDAセットアップ回路141においては、まず三角形プリミティブPM101に係るポリゴンレンダリングデータS11に基づいて、三角形の辺と水平方向の差分などを示す変分データが生成される。具体的には、開始点の値と終点の値、並びに、その間の距離を用いて、単位長さ移動した場合における、求めようとしている値の変化分である変分が算出され、変分データを含む三角形プリミティブPM101に関する情報が、図17に示すように、ブロードキャストバス149を通して、各処理モジュール140-0~140-3の領域判定回路142-0~142-3に並列に供給される。

【0111】各処理モジュール140-0～140-3の領域判定回路142-0～142-3においては、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形プリミティブPM101に関する情報を受けて、三角形があらかじめ規定されている自モジュールの担当領域RGN～RGN3に含まれるか否かが判定される。この場合、上述したように、三角形プリミティブPM101は、全処理モジュール140-0～140-3の担当領域RGN0～RGN3に含まれることから、図18に示すよう

【0112】これと並行として、次の三角形プリミティブPM102に係るポリゴンレンダリングデータS11がDDAセットアップ回路141に入力される。DDAセットアップ回路141においては、三角形プリミティブPM102に係るポリゴンレンダリングデータS11に基づいて、三角形の辺と水平方向の差分などを示す変分データが生成される。具体的には、開始点の値と終点の値、並びに、その間の距離を用いて、単位長さ移動した場合における、求めようとしている値の変化分である変分が算出され、変分データを含む三角形プリミティブPM102に関する情報が、図18に示すように、ブロードキャストバス149を通して、各処理モジュール140-0～140-3の領域判定回路142-0～142-3に並列に供給される。

【0113】各処理モジュール140-0～140-3の領域判定回路142-0～142-3においては、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形プリミティブPM102に関する情報を受けて、三角形があらかじめ規定されている自モジュールの担当領域RGN～RGN3に含まれるか否かが判定される。この場合、上述したように、三角形プリミティブPM102は、処理モジュール140-0、140-1の担当領域RGN0、RGN1に含まれ、処理モジュール140-2、140-3の担当領域RGN2、RGN3に含まれない。したがって、図19に示すように、担当領域RGN0、RGN1に含まれると判定した各領域判定回路142-0、142-1から入力情報が自モジュールのDDA回路143-0、143-1に出力される。一方、担当領域RGN2、RGN3に含まれないと判定した処理モジュール140-2、140-3の領域判定回路142-2、142-3では、入力情報が次段のDDA回路143-2、143-3に出力されずに棄てられる。

【0114】これと並行として、次の三角形プリミティブPM103に係るポリゴンレンダリングデータS11がDDAセットアップ回路141に入力される。DDAセットアップ回路141においては、三角形プリミ

ブPM103に係るポリゴンレンダリングデータS11に基づいて、三角形の辺と水平方向の差分などを示す変分データが生成される。具体的には、開始点の値と終点の値、並びに、その間の距離を用いて、単位長さ移動した場合における、求めようとしている値の変化分である変分が算出され、変分データを含む三角形プリミティブPM103に関する情報が、図19に示すように、ブロードキャストバス149を通して、各処理モジュール140-0～140-3の領域判定回路142-0～142-3に並列に供給される。

【0115】各処理モジュール140-0～140-3の領域判定回路142-0～142-3においては、ブロードキャストバス149を介して入力したDDAセットアップ回路141による変分データを含む三角形プリミティブPM103に関する情報を受けて、三角形があらかじめ規定されている自モジュールの担当領域RGN～RGN3に含まれるか否かが判定される。この場合、上述したように、三角形プリミティブPM103は、全処理モジュール140-0～140-3の担当領域RGN0～RGN3に含まれることから、図20に示すように、各領域判定回路142-0～142-3から入力情報が自モジュールのDDA回路143-0～143-3に出力される。

【0116】各処理モジュール140-0～140-3のDDA回路143-0～143-3では、三角形プリミティブPM101～PM103の入力順で以下の処理が行われる。すなわち、まず、全DDA回路143-0～143-3において、三角形プリミティブPM101に対する処理が行われ、次に、処理モジュール140-0、140-1のDDA回路143-0、143-1において、三角形プリミティブPM102に対する処理が行われ、最後に、全DDA回路143-0～143-3において、三角形プリミティブPM103に対する処理が行われる。同様に、DDA回路143-0～143-3に後続するテクスチャエンジン回路144-0～144-3、さらにメモリI/F回路145-0～145-3では、三角形プリミティブPM101～PM103の入力順で以下の処理が行われる。

【0117】具体的には、全DDA回路143-0～143-3またはDDA回路143-0および143-1においては、それぞれ領域判定回路142-0～142-3から供給された変分データを用いて、三角形内部の各ピクセルにおける線形補間された(z, R, G, B, s, t, q)データが算出される。そして、算出された(z, R, G, B, s, t, q)データと、三角形の各頂点の(x, y)データとが、DDAデータとして、テクスチャエンジン回路144-0～144-3に出力される。

【0118】テクスチャエンジン回路144-0～144-3またはテクスチャエンジン回路144-0および

10

20

30

40

50

144-1においては、DDA回路143-0~143-3によるDDAデータが示す(s, t, q)データについて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とが行われる。そして、除算結果「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEが乗算され、テクスチャ座標データ(u, v)が生成される。

【0119】次に、テクスチャエンジン回路144-0~144-3またはテクスチャエンジン回路144-0および144-1から専用メモリI/F回路146に対して生成されたテクスチャ座標データ(u, v)を含む読み出し要求が出力され、専用メモリI/F回路146を介して、グラフィックスメモリ147に記憶された

(R, G, B)データが読み出される。次に、テクスチャエンジン回路144-0~144-3またはテクスチャエンジン回路144-0および144-1において、読み出した(R, G, B)データと、前段のDDA回路143-0~143-3またはDDA回路143-0および143-1からのDDAデータに含まれる(R, G, B)データとが掛け合わされ、ピクセルデータとして生成される。このピクセルデータは、テクスチャエンジン回路144-0~144-3またはテクスチャエンジン回路144-0および144-1からメモリI/F回路145-0~145-3またはメモリI/F回路145-0および145-1に出力される。

【0120】そして、メモリI/F回路145-0~145-3またはメモリI/F回路145-0および145-1において、テクスチャエンジン回路144-0~144-3またはテクスチャエンジン回路144-0および144-1から入力したピクセルデータに対応するzデータと、対応するメモリモジュール147-0~147-3またはメモリモジュール147-0および147-1(zバッファ)に記憶されているzデータとの比較が行われ、入力したピクセルデータによって描画される画像が、前回、メモリモジュールのディスプレイバッファに書き込まれた画像より、手前(視点側)に位置するか否かが判定される。判定の結果、手前に位置する場合には、画像データに対応するzデータでメモリモジュール147-0~147-3またはメモリモジュール147-0および147-1に記憶されたzデータが更新される。

【0121】次に、メモリI/F回路145-0~145-3またはメモリI/F回路145-0および145-1において、(R, G, B)データが対応する自モジュールのメモリモジュール147-0~147-3またはメモリモジュール147-0および147-1に書き込まれる。

【0122】また、専用メモリI/F回路146においては、今から描画しようとしているピクセルにおけるテクスチャアドレスに対応したテクスチャを格納している

メモリブロックがそのテクスチャアドレスにより算出され、グラフィックスメモリ147の所定のメモリモジュールにのみ読みだし要求が出され、テクスチャデータが読み出される。

【0123】また、メモリI/F回路145-0~145-3またはメモリI/F回路145-0および145-1において、今から描画しようとしているピクセルアドレスに対応するピクセルデータを格納しているメモリモジュール147-0~147-3またはメモリモジュール147-0および147-1に対して、該当アドレスからピクセルデータがモディファイ書き込み(Modify Write)を行うために読み出され、モディファイ後、同じアドレスへ書き戻される。

【0124】メモリI/F回路145-0~145-3またはメモリI/F回路145-0および145-1において、隠れ面処理を行う場合には、やはり同じように今から描画しようとしているピクセルアドレスに対応する奥行きデータを格納しているメモリモジュール147-0~147-3またはメモリモジュール147-0および147-1に対して、該当アドレスから奥行きデータがモディファイ書き込み(Modify Write)を行うために読み出され、必要ならばモディファイ後、同じアドレスへ書き戻される。

【0125】そして、図示しないCRTに画像を表示する場合には、CRTコントロール回路148において、与えられた水平垂直同期周波数に同期して、表示アドレスが発生され、専用メモリI/F回路146へ表示データ転送の要求が出される。専用メモリI/F回路146では、その要求に従い、一定のまとまった固まりで、表示データがCRTコントロール回路148に転送される。CRTコントロール回路148では、図示しないディスプレイ用FIFO等にその表示データが貯えられ、一定の間隔でRGBのインデックス値が生成される。CRTコントロール回路148においては、RAM内部にRGBのインデックスに対するRGB値が記憶されていて、インデックス値に対するRGB値が図示しないD/Aコンバータへ転送される。そして、D/Aコンバータでアナログ信号に変換されたRGB信号がCRTへ転送される。

【0126】以上説明したように、本実施形態によれば、DDAセットアップ回路141による同じ三角形データを各処理モジュール140-0~140-3に並列に送り、各処理モジュール140-0~140-3は、各メモリモジュール147-0~147-3と対応付けられており、各処理モジュール140-0~140-3は、DDAセットアップ回路141による変分データを含む三角形に関する情報を受けて、三角形があらかじめ規定されている自モジュールの担当領域RGN0~RGN3に含まれるか否かを判定し、含まれる場合には、入力情報を次段のDDA回路143-0~143-3に出

力し、含まれていない場合には、入力情報を次段のDDA回路143-0~143-3に出力せずに棄てる領域判定回路142-0~142-3を有し、自モジュールの担当する領域RGN0~RGN3に含まれるピクセルのみを描画して対応するメモリモジュール147-0~147-3に送るようにしたので、複数の処理モジュールが処理データを共有して並列処理する際に、動作モードにかかわらず並列処理を行うことができる。そして、並列処理できるピクセル数に対する制約を受けずに実行効率の高い並列処理を実現でき、ひいては、性能の向上を図ることができる利点がある。また、本実施形態では、各処理モジュール140-0~140-3とメモリモジュール147-0~147-3を直結できる処理分配とし、配線本数を削減し、小型化することができる。その結果、設計が容易となり、配線コスト、配線遅延を低減することができる。

【0127】なお、上述した実施形態においては、図13(A)~(E)に示すように、各処理モジュールの担当領域の大きさとDDA回路がピクセルを発生する矩形領域の大きさとが一致している場合を例に説明したが、本発明はこれに限定されるものではない。

【0128】すなわち、担当領域の大きさと矩形領域の大きさは一致している必要はない。たとえば、図21(A)~(E)に示すように、各担当領域RGN0~RGN3においてピクセルを発生する矩形領域を2×2で階層化することも可能である。この場合、移動は階層的になる。すなわち、担当領域間での移動と担当領域内での矩形領域単位での移動が行われる。そして、各処理モジュールにおいては、複数の単位領域における担当領域をスキャンし、階層的に分割領域単位で同時にピクセルを発生する。この場合も、上述した実施形態と同様の効果を得ることができる。

【0129】また、上述した実施形態においては、図13(A)~(E)に示すように、各処理モジュール140-0~140-3の担当領域RGN0~RGN3の割り付けを固定、具体的には、2×2の矩形領域の左上を処理モジュール140-0の担当領域とし、右上を処理モジュール140-1の担当領域とし、左下を処理モジュール140-2の担当領域とし、右下を処理モジュール140-3の担当領域とした場合を例に説明したが、本発明はこれに限定されるものではない。

【0130】すなわち、担当領域の割り付けは任意に行うことも可能である。たとえば、図22(B)に示すように、左上および左下の2×2の矩形領域においては、上述したように、左上を処理モジュール140-0の担当領域とし、右上を処理モジュール140-1の担当領域とし、左下を処理モジュール140-2の担当領域とし、右下を処理モジュール140-3の担当領域とし、右上および右下の2×2の矩形領域においては、左下を処理モジュール140-0の担当領域とし、右下を処理

モジュール140-1の担当領域とし、右上を処理モジュール140-2の担当領域とし、右上を処理モジュール140-3の担当領域とする等、種々の態様が可能である。

【0131】担当領域間での移動は、図22(C)~(F)に示すように行われる。処理モジュール140-0のDDA回路143-0は、上述したように、2×2の矩形領域の左上領域および左下領域が担当領域である。本例では、図22(C)に示すように、破線で示す処理開始点が自モジュールの担当領域RGN0であることから、DDA回路143-0は、次に、右斜め上方に移動し、次に、左斜め上方に移動する。

【0132】処理モジュール140-1のDDA回路143-1は、上述したように、2×2の矩形領域の右上領域および右下領域が担当領域である。本例では、図22(D)に示すように、破線で示す処理開始点からX方向にスキャンして自モジュールの担当領域RGN1に移動する。そして、DDA回路143-1は、右斜め上方に移動し、次に、左斜め上方に移動する。

【0133】処理モジュール140-2のDDA回路143-2は、上述したように、2×2の矩形領域の左下領域および左上領域が担当領域である。本例では、図22(E)に示すように、破線で示す処理開始点からY方向(図面で上方向)にスキャンして自モジュールの担当領域RGN2に移動する。そして、DDA回路143-2は、次に、右斜め上方に移動する。

【0134】処理モジュール140-3のDDA回路143-3は、上述したように、2×2の矩形領域の右下領域および右上領域が担当領域である。本例では、図22(F)に示すように、破線で示す処理開始点からX方向にスキャンしてさらにY方向にスキャンして自モジュールの担当領域RGN3に移動する。そして、DDA回路143-3は、次に、右斜め上方に移動する。

【0135】この場合も、上述した実施形態と同様の効果を得ることができる。

【0136】また、上述した実施形態においては、同一のプリミティブに係る担当領域を複数の処理モジュール140-0~140-3に割り当てて並列に処理するように構成したが、たとえば、複数のプリミティブを同時に処理するように構成することも可能である。具体的には、複数の処理モジュールに別々のプリミティブ(たとえば三角形)を与えて並列動作させることも可能である。

【0137】図23は、本発明に係る画像処理装置の他の実施形態を説明するための図で、プリミティブレベルでの並列化処理について概念的に示す図である。図23の画像処理装置(図3および図4の対応ではレンダリング回路)14Aは、並列に配置された処理モジュール140A-0~140A-3の前段に単位図形(プリミティブ)を各処理モジュール140A-0~140A-3

10

20

30

40

50

に分配する単位図形分配回路 151 を設けている。

【0138】この実施形態では、各処理モジュール 140A-0~140A-3 に比較的均等大きさのプリミティブが与えられているときには、各処理モジュール 140A-0~140A-3 に対する負荷のバランスがとれ、効率的並列処理が行える。また、この実施形態では、グラフィックス処理を行うハードウェアでは、メモリのバンド幅を広げるために、複数メモリモジュールを用いてメモリインターリーブを行うことから、各処理モジュール 140A-0~140A-3 と各メモリモジュール 147-0~147-3 を全て結ぶ必要がある。そのため、クロスバー回路 152 が設けられている。

【0139】図 23 の回路によれば、配線数、配線コストが増えることになるが、複数の処理モジュールが処理データを共有して並列処理する際に、動作モードにかかわらずに並列処理を行うことができる。そして、並列処理できるピクセル数に対する制約を受けずに実行効率の高い並列処理を実現でき、ひいては、性能の向上を図ることができる利点がある。

【0140】なお、図 3 に示す 3 次元コンピュータグラフィックスシステム 10 では、ポリゴンレンダリングデータを生成するジオメトリ処理を、メインプロセッサ 11 で行う場合を例示したが、レンダリング回路 14 で行う構成にしてもよい。

【0141】

【発明の効果】以上説明したように、本発明によれば、複数の処理装置が処理データを共有して並列処理する際に、動作モードにかかわらずに並列処理を行うことができる。そして、本発明によれば、並列処理できるピクセル数に対する制約を受けずに実行効率の高い並列処理を実現でき、ひいては、性能の向上を図ることができる利点がある。

【図面の簡単な説明】

【図 1】グラフィックス描画を行う一般的な画像処理装置を概念的に示す図である。

【図 2】ピクセルレベルでの並列処理の手法に基づくプリミティブレベルでの並列化処理について概念的に示す図である。

【図 3】本発明に係る画像処理装置としての 3 次元コンピュータグラフィックスシステムのシステム構成図である。

【図 4】本実施形態に係るレンダリング回路の一実施形態を示すブロック構成図である。

【図 5】本実施形態に係る並列化処理の基本概念を説明するための図である。

【図 6】本実施形態に係る並列化処理の基本概念を説明するための図である。

【図 7】本実施形態に係る DDA セットアップ回路の機能を説明するための図である。

【図 8】本実施形態に係る DDA セットアップ回路の構

成例を示す図である。

【図 9】本実施形態に係る処理モジュールの担当領域について説明するための図である。

【図 10】本実施形態に係る領域判定回路の判定処理を概念的に説明するための図であって、4 並列の場合の例を示す図である。

【図 11】本実施形態に係る領域判定回路の判定処理を概念的に説明するための図であって、8 並列の場合の例を示す図である。

【図 12】本実施形態に係る領域判定回路の判定処理動作を説明するためのフローチャートである。

【図 13】本実施形態に係る DDA 回路のピクセル生成処理を説明するための図である。

【図 14】本実施形態に係る DDA 回路のピクセル生成処理動作を説明するためのフローチャートである。

【図 15】図 4 の回路の動作を説明するための図であって、連続的に処理される三角形プリミティブの例を示す図である。

【図 16】図 4 の回路の動作を説明するための図である。

【図 17】図 4 の回路の動作を説明するための図である。

【図 18】図 4 の回路の動作を説明するための図である。

【図 19】図 4 の回路の動作を説明するための図である。

【図 20】図 4 の回路の動作を説明するための図である。

【図 21】各処理モジュールの担当領域の大きさと DDA 回路がピクセルを発生する矩形領域の大きさとが一致していない場合の例を説明するための図である。

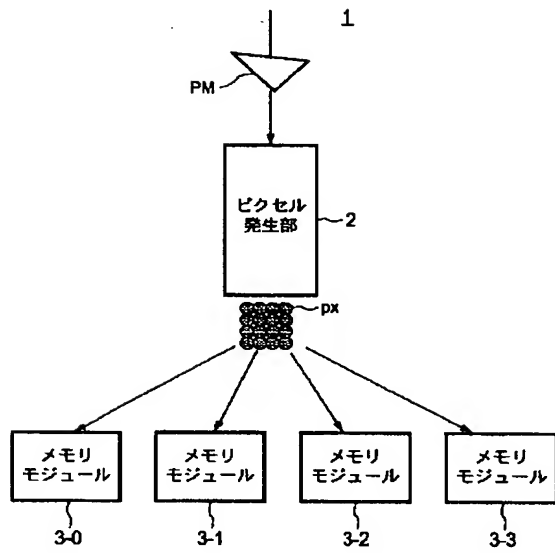
【図 22】各処理モジュールの担当領域の割り付けの他の例を説明するための図である。

【図 23】本発明に係る画像処理装置の他の実施形態を説明するための図である。

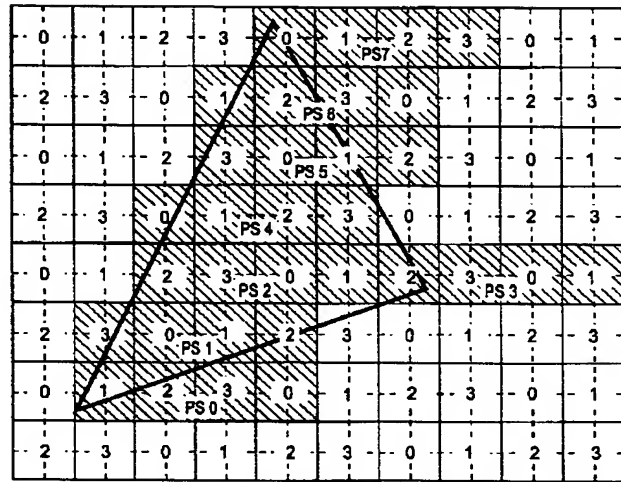
【符号の説明】

10…画像処理装置、11…メインプロセッサ、12…メインメモリ、13…I/O インタフェース回路、14、14A…レンダリング回路、140-0~140-3、140A-0~140A-3…処理モジュール、142-0~142-3…領域判定回路、143-0~143-3…トライアングル DDA 回路、144-0~144-3…テクスチャエンジン回路、145-0~145-3…メモリインタフェース (I/F) 回路、146…専用メモリ I/F 回路、147…グラフィックスメモリ、147-0~147-3…メモリモジュール、148…CRT コントロール回路、149…セットアップデータ用ブロードキャストバス、150…テクスチャバス、151…単位図形分配回路、152…クロスバー回路、RGN0~RGN3…担当領域。

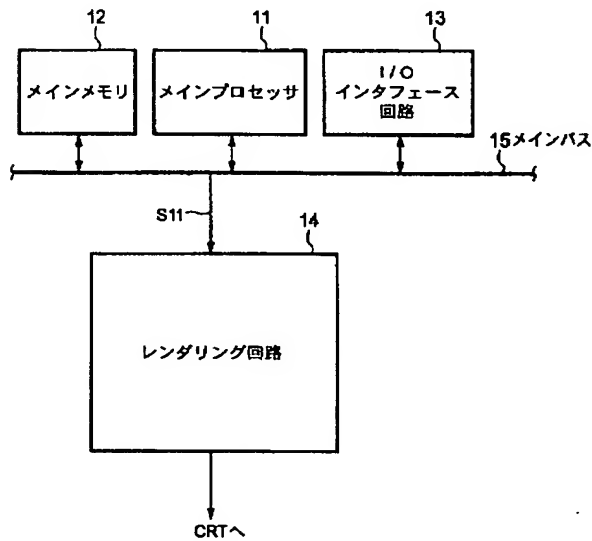
【図 1】



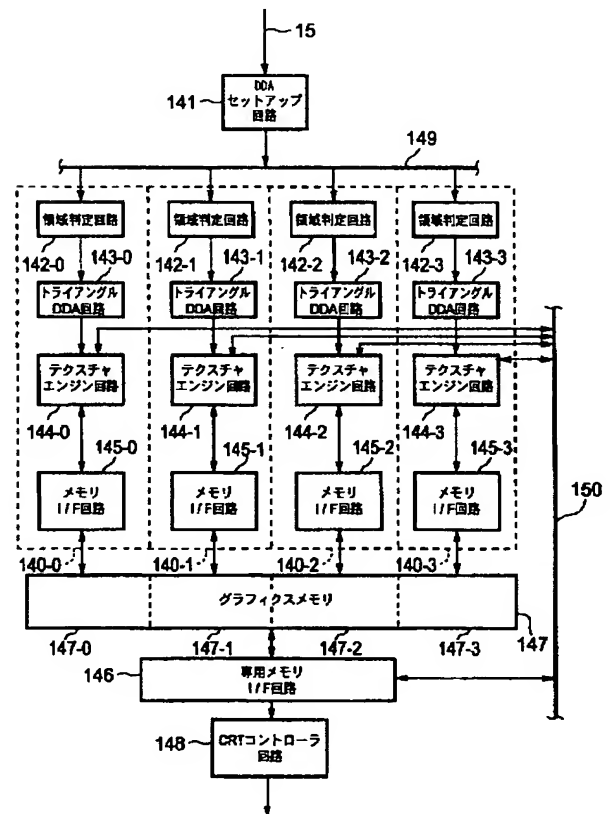
【図 2】



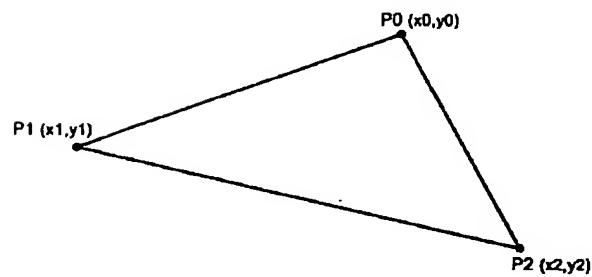
【図 3】



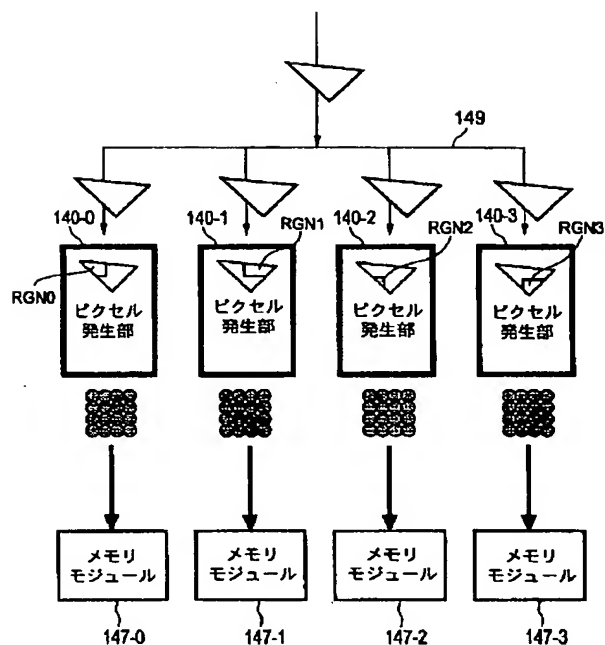
【図 4】



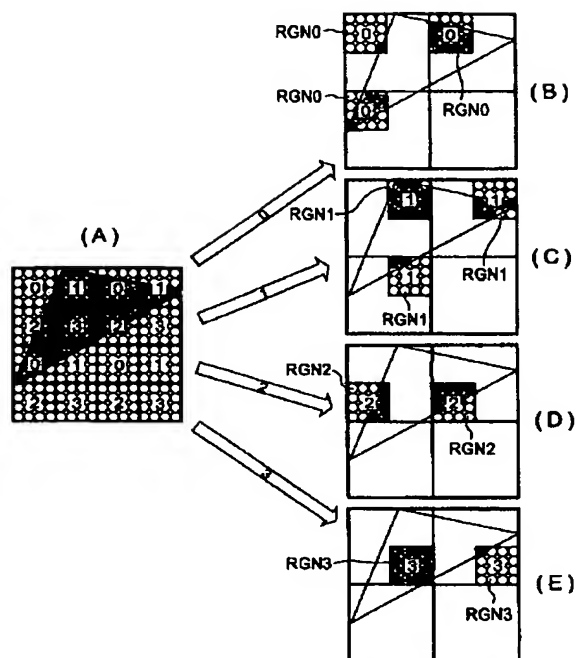
【図 7】



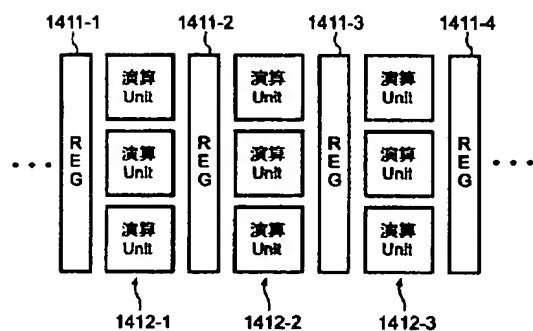
【図5】



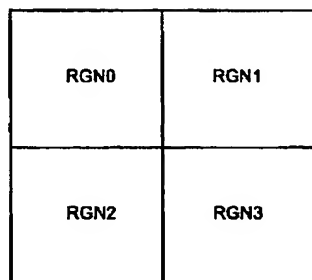
【図6】



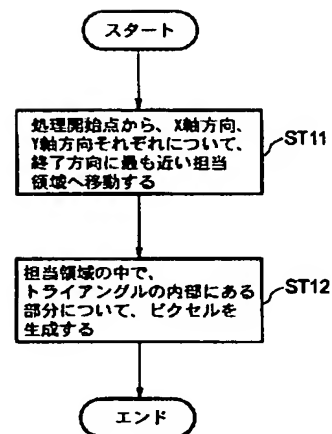
【図8】



【図9】

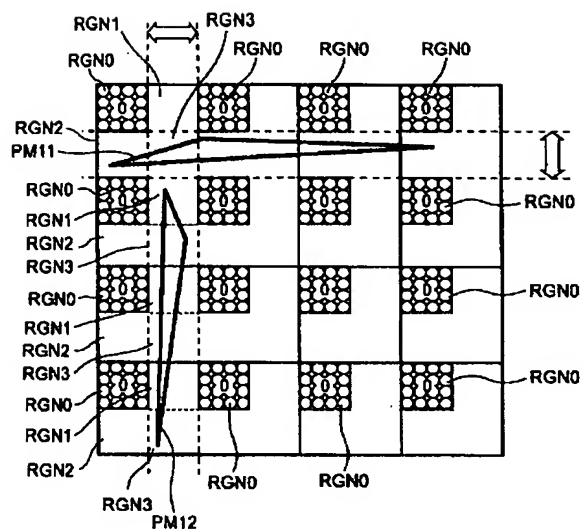


【図14】

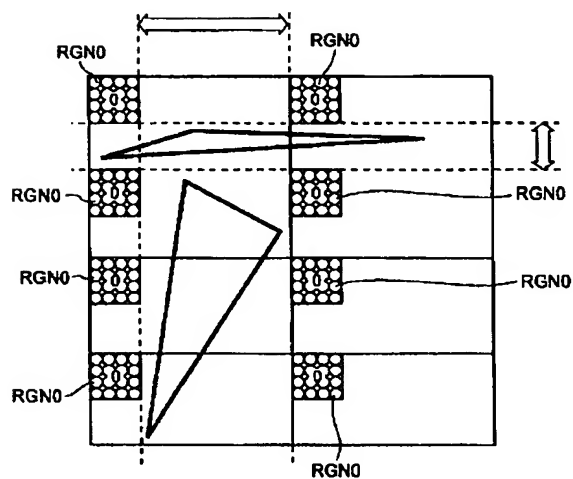




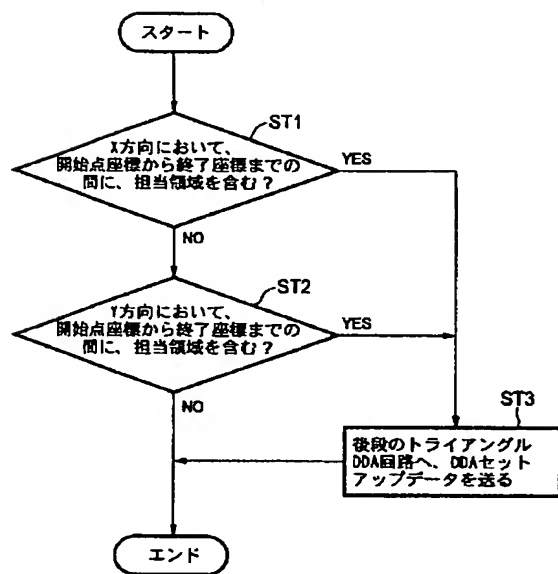
【図10】



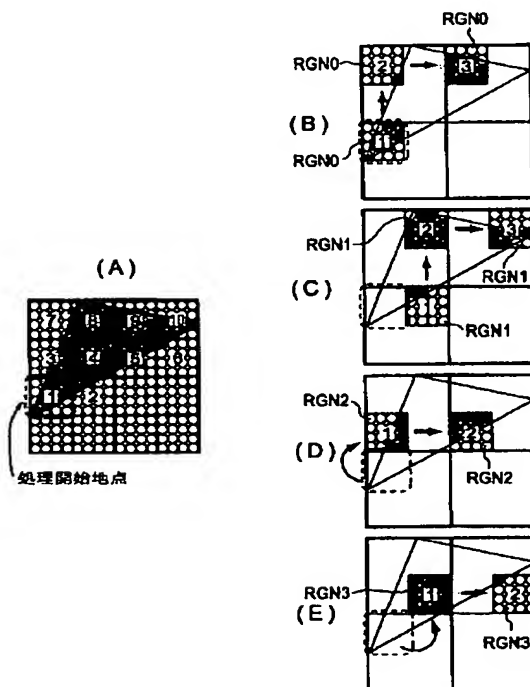
【図11】



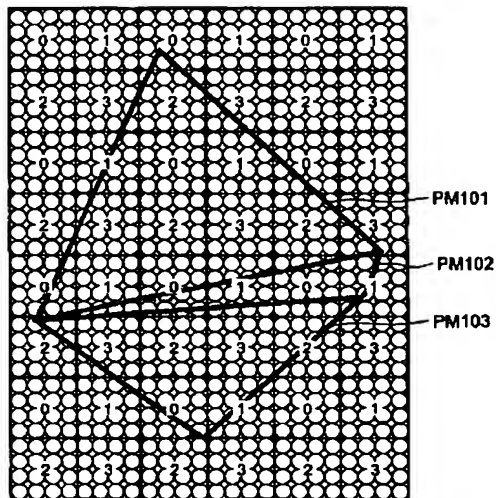
【図12】



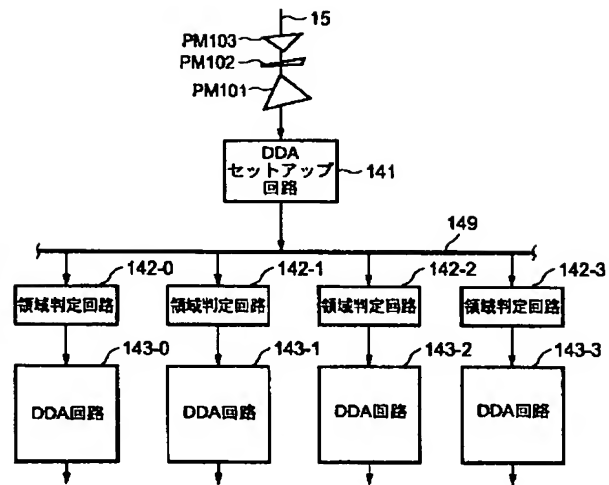
【図13】



【図15】

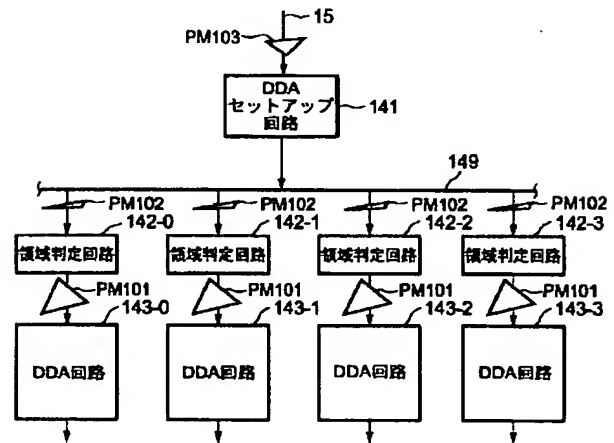
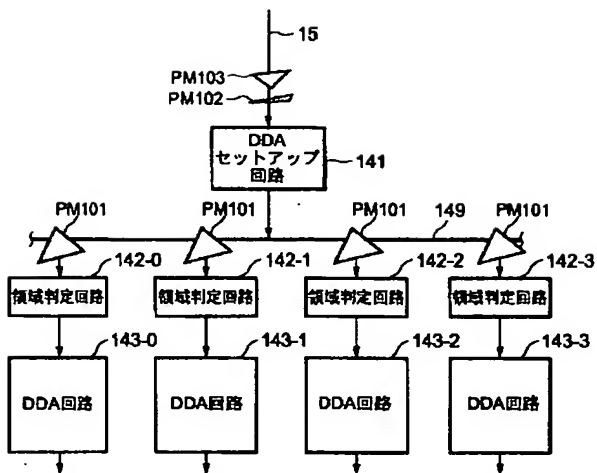


【図16】

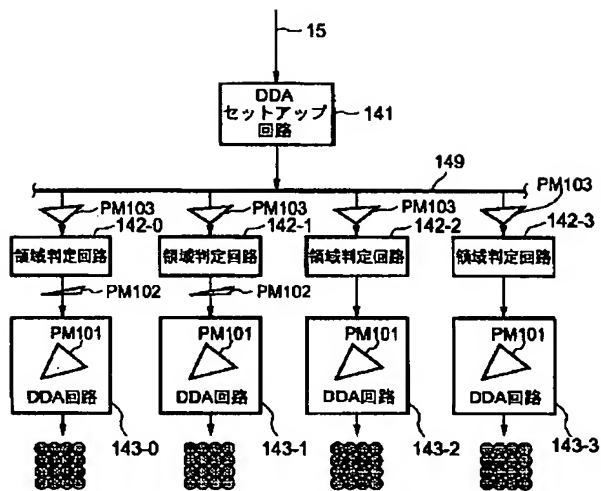


【図18】

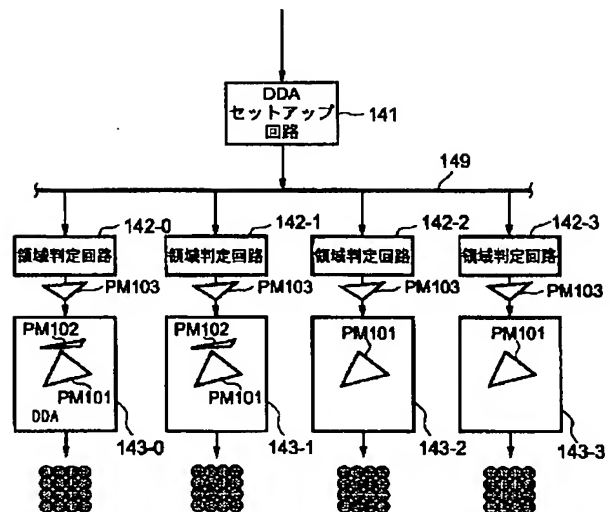
【図17】



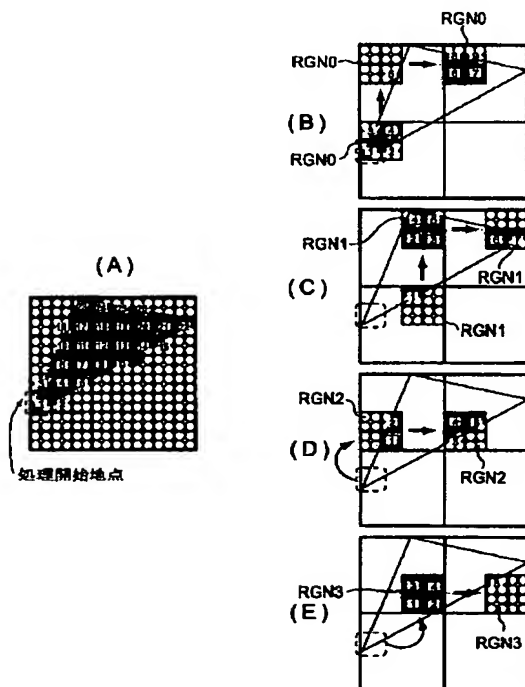
【図 19】



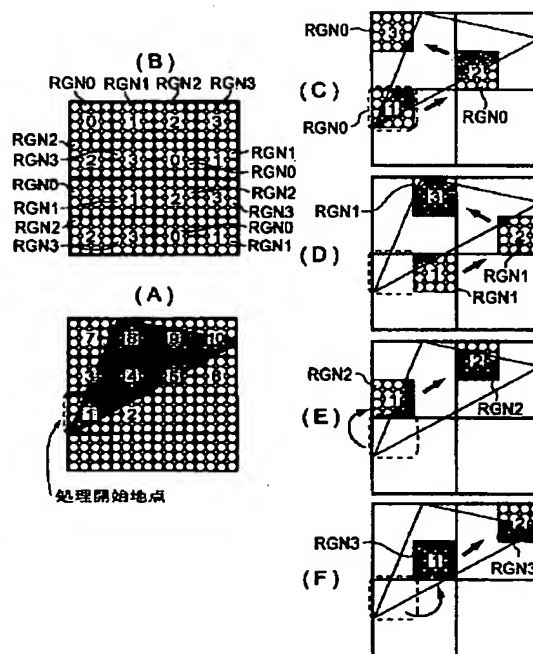
【図 20】



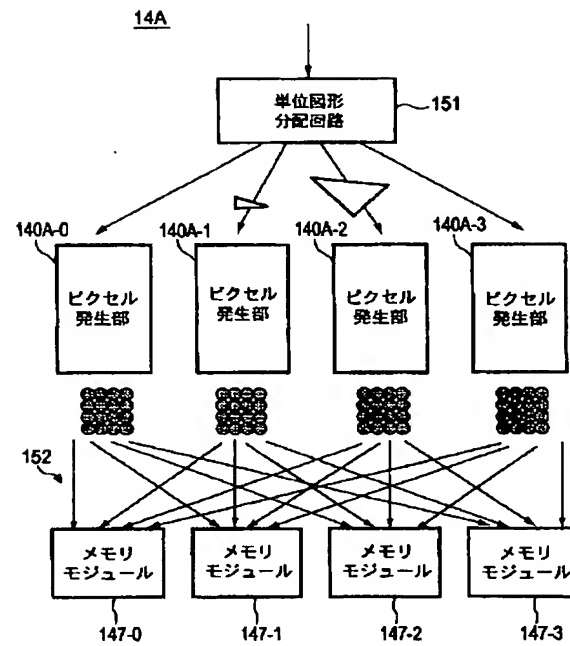
【図 21】



【図 22】



【図 23】



フロントページの続き

(72)発明者 山口 裕司  
東京都品川区東五反田 1 丁目14番10号 株  
式会社ソニー木原研究所内

Fターム(参考) 5B045 DD03 GG11  
5B057 AA20 CA01 CA08 CA12 CA13  
CA17 CB01 CB08 CB12 CB16  
CC04 CH04 CH11  
5B080 AA14 CA03 CA09 FA02 GA22